



**Tiago José de
Figueiredo Lopes**

**Solução inovadora para monitorização e controlo
no âmbito da domótica**



**Tiago José de
Figueiredo Lopes**

**Solução inovadora para monitorização e controlo
no âmbito da domótica**

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestrado em Engenharia Mecânica, realizada sob orientação científica de José Paulo Oliveira Santos, Professor Auxiliar do Departamento de Engenharia Mecânica da Universidade de Aveiro.

O júri / The jury

Presidente / President

Prof. Doutor Vítor Manuel Ferreira dos Santos

Professor Associado da Universidade de Aveiro

Vogais / Committee

Prof. Doutor Arnaldo Silva Rodrigues de Oliveira

Professor Auxiliar da Universidade de Aveiro (arguente)

Prof. Doutor José Paulo Oliveira Santos

Professor Auxiliar da Universidade de Aveiro (orientador)

**Agradecimentos /
Acknowledgements**

Desejo agradecer ao meu orientador Prof. José Paulo Santos, pela disponibilidade, atenção dispensada, paciência e profissionalismo.

Agradeço à minha família por todo o apoio dado ao longo desta etapa e no decorrer de todo o curso.

Agradeço aos amigos pela amizade e por todo o apoio e motivação.

Agradeço-te a ti ...

Apoio financeiro dos projetos UID/EMS/00481/2013-FCT e CENTRO-01-0145-FEDER-022083.

Palavras-chave

Domótica; *Esp8266*; Aquecimento de ar; Aquecimento de águas, Painel solar; Interface *web*; *Wi-fi*; Algoritmos; Energia; Automação.

Resumo

O avanço da tecnologia tem influenciado positivamente a domótica, tendo contribuído bastante para o desenvolvimento de sistemas mais ecológicos, mais eficientes, mais intuitivos e com custos mais reduzidos.

Atualmente o mercado oferece alguns sistemas de controlo e monitorização de variáveis nas habitações, estes sistemas são abordados no início da presente dissertação, assim como alguns algoritmos inteligentes que podem ser aplicados nestes sistemas. Também são abordadas várias soluções de aquecimento de águas sanitárias.

O trabalho desenvolvido consiste em criar sistemas que possuam a capacidade de controlar e monitorizar várias grandezas em habitações genéricas. Estes sistemas têm de ser capazes de monitorizar e controlar grandezas via *web*.

Desenvolveram-se três sistemas distintos e uma interface *web* ao longo da dissertação. Estes sistemas focam-se no aquecimento de ar e água e na rotação de um painel solar. Todos os sistemas desenvolvidos têm por base o menor consumo de energia.

Após a análise de desempenho realizada aos sistemas desenvolvidos conclui-se que estes são viáveis de implementar em realidade habitacional, no entanto mais trabalho de desenvolvimento do produto terá de ser realizado para se desenvolver uma solução final.

Keywords

Domotics; *Esp8266*; Air heating; Water heating; Solar panel; *Web* interface; *Wi-fi*; Algorithms; Energy; Automation.

Abstract

The development of technology had positively influenced the field of domotics, had contributed a lot for new solutions which are more ecological, efficient, intuitive and with decreased costs.

Nowadays the market already offers some control and monitoring systems for home solutions. These ones are addressed in the beginning of this dissertation as well as intelligent algorithms. A number of solutions of warm systems of sanitary waters are also presented.

The work developed in the present dissertation was focused on creating a solution which had the capability of controlling and monitoring several variables in generic houses. These solution have to have the ability to be monitored and controlled via web.

It had been developed 3 distinct systems and one interface along the dissertation. These systems focused on air and water heating and on the rotation of a solar panel.

After the performance analysis made on the developed solution it can be concluded that those would be feasible to implement on housing reality, however more research and development would have to be made to develop a final product.

Conteúdo

1	Introdução	1
1.1	Enquadramento	1
1.2	Motivação	2
1.3	Objetivos	3
1.4	Estrutura do documento	3
2	Conceitos Fundamentais e Estado da Arte	5
2.1	Fenómenos de Transferência de Calor	5
2.1.1	Condução	5
2.1.2	Convecção	5
2.2	Algoritmos	6
2.2.1	<i>Fuzzy Logic</i>	7
2.3	Principais soluções de domótica	7
2.3.1	KNX - Konnex	7
2.3.2	X10	7
2.4	Sistemas de Águas Quentes Sanitárias existentes	8
2.5	<i>Controlling the average residential electric water heater power demand using fuzzy logic</i>	12
2.6	<i>Development of Automatisms and intelligent control algorithms, for monitoring and control of housing components</i>	15
3	Arquitetura da Solução Proposta	19
3.1	Solução proposta	19
3.1.1	Arquitetura da solução proposta	20
3.1.2	Servidor	21
3.1.3	Cliente	22
3.2	Interações na Solução Proposta	24
4	Implementação	29
4.1	Sistema Proposto	29
4.2	<i>Hardware</i>	33
4.2.1	<i>Esp8266</i>	33
4.2.2	Placa de relés	34
4.2.3	Sistema Água - <i>Hardware</i>	34
4.2.4	Sistema Motor - <i>Hardware</i>	35
4.2.5	Sistema Ar - <i>Hardware</i>	36
4.2.6	Considerações gerais sobre o sistema proposto	37

4.3	<i>Software</i>	37
4.3.1	Desenvolvimento Comum	38
4.3.2	Sistema Água - <i>Software</i>	40
4.3.3	Sistema Motor - <i>Software</i>	43
4.3.4	Sistema Ar - <i>Software</i>	45
4.3.5	Servidor - <i>Software</i>	47
5	Análise de Desempenho	51
5.1	Casos de Estudo / Protótipos Laboratoriais	51
5.2	Análise de Desempenho dos casos de Estudo	54
6	Conclusão	59
6.1	Trabalho Futuro	60
A	Circuitos do Sistema Proposto	63
B	Interface <i>web</i>	67
B.1	Organização	67
B.1.1	Menu	68
B.1.2	Área de Trabalho	68
B.2	Considerações gerais	69

Lista de Tabelas

2.1	Valores obtidos pelo autor nos vários testes que realizou.	16
4.1	Tabela Solar térmico	49
4.2	Tabela Termo acumulador	49
4.3	Tabela Movimento painel	49
4.4	Tabela Ar interior	50
4.5	Tabela Histórico termo acumulador	50
4.6	Tabela Histórico ar	50

Lista de Figuras

1.1	Distribuição do consumo doméstico de Eletricidade por tipo de utilização - Portugal, 2010 (Adaptado de [1])	2
2.1	Condução de calor num sólido com duas faces a temperaturas diferentes (Adaptado de [2])	6
2.2	Convecção de uma superfície para um fluido em movimento (Adaptado de [2])	6
2.3	Diferenças entre lógica difusa e lógica booleana [3]	7
2.4	Esquema de comunicação do protocolo X10 (Adaptado de [4])	8
2.5	Esquema de funcionamento de um esquentador. (Adaptado de [5])	9
2.6	Esquema de um termoacumulador elétrico. (Adaptado de [6])	9
2.7	Esquema de funcionamento de um sistema solar com tecnologia de Termossifão. (Adaptado de [7])	10
2.8	Esquema de funcionamento de um sistema solar com tecnologia de Circulação Forçada. (Adaptado de [7])	10
2.9	Esquema que ilustra o princípio de funcionamento de uma bomba de calor. (Adaptado de [8])	11
2.10	Esquema de sistemas com recurso a mais do que uma fonte de energia para aquecimento de águas sanitárias. (Adaptado de [9])	12
2.11	Esquema de um sistema que recorre a múltiplas fontes de energia para o aquecimento de águas sanitárias. (Adaptado de [10])	12
2.12	Potência despendida no aquecimento de águas sanitárias e no total em função da hora ao longo do dia [11]	13
2.13	Diagrama de blocos do controlador implementado com <i>Fuzzy Logic</i> [11].	14
2.14	Comparação de um equipamento controlado e um não controlado [11]	14
2.15	Comparação entre dois equipamentos controlados e dois equipamentos em que um é controlado e o outro não [11].	15
2.16	Comparação da implementação dos vários algoritmos [3].	16
2.17	Representação da arquitetura do modelo construído pelo autor [3].	17
2.18	Esquema da solução implementada pelo autor [3].	18
3.1	Habitação genérica com os vários sistemas prontos a monitorizar e controlar.	20
3.2	Modelo Cliente/Servidor aplicado à solução proposta.	21
3.3	Arquitetura da solução proposta.	21
3.4	Elementos constituintes do servidor.	22
3.5	Elementos que constituem o cliente externo	23
3.6	Elementos que constituem o cliente controlador	23

3.7	Interações entre os intervenientes da solução proposta.	24
3.8	Interação do cliente externo com o servidor.	25
3.9	Mensagens trocadas entre o cliente externo e o servidor.	25
3.10	Interação do cliente controlador com o servidor.	26
3.11	Mensagens trocadas entre o cliente controlador e o servidor.	26
3.12	Mensagens trocadas entre o Servidor <i>web</i> e o Servidor Base de Dados . . .	27
4.1	Planta genérica de uma habitação.	29
4.2	Sistema de monitorização e controlo de AQS acoplado com um cliente controlador	30
4.3	Sistema de monitorização e controlo da rotação de um painel solar acoplado com um cliente controlador	30
4.4	Sistema de monitorização e controlo do ar interior acoplado com um cliente controlador	31
4.5	Planta genérica de uma habitação com os sistemas propostos instalados. .	31
4.6	Planta genérica com os sistemas instalados, o servidor , o meio de comunicação e o cliente externo	32
4.7	Várias janelas da interface <i>web</i>	32
4.8	<i>Hardware</i> base do cliente controlador	33
4.9	Placa de desenvolvimento que contém o <i>chip Esp8266 12F - Esp8266</i> MINI NODEMCU.	33
4.10	Placa de 4 relés [12].	34
4.11	Alguns componentes utilizados no sistema água. A -LM35 B -YF-S401 C -ADS1115	35
4.12	Alguns componentes utilizados no sistema motor. A -Redutor B -Variador C -Encoder D -Sensor Indutivo	36
4.13	Optoacopladores. A -4N25 B -IL300	37
4.14	Janela do Visual Studio.	38
4.15	Organização das camadas do protocolo TCP/IP [13].	38
4.16	Código que estabelece a ligação do <i>Esp8266</i> ao <i>router</i>	39
4.17	Função que envia um pedido de página PHP ao servidor, com o valor atualizado das variáveis a monitorizar.	39
4.18	Função que realiza a leitura e interpreta a resposta dada pela base de dados, através do ficheiro PHP.	40
4.19	Algoritmo desenvolvido para o modo automático do sistema água.	41
4.20	Diagrama do algoritmo desenvolvido para o sistema água.	42
4.21	Algoritmo desenvolvido para o modo automático do sistema motor.	43
4.22	Diagrama do algoritmo desenvolvido para o sistema motor.	44
4.23	Deteção de um sensor de fim de curso em modo automático	45
4.24	Algoritmo desenvolvido para o modo automático do sistema motor.	46
4.25	Esquema ilustrativo da lógica utilizada para desenvolver o algoritmo do sistema ar.	47
4.26	Interface do <i>software Xampp</i>	48
4.27	Interface <i>phpMyAdmin</i>	48
5.1	Caso de estudo para o sistema água.	51
5.2	Caso de estudo para o sistema motor.	52

5.3	Caso de estudo para o sistema ar.	53
5.4	Várias janelas da interface desenvolvida para controlo e monitorização. . .	53
5.5	Comparação do valor medido com o LM35 conectado à ADC do <i>Esp8266</i> e com o sensor do multímetro.	54
5.6	Comparação do valor medido com o LM35 conectado à placa ADS 1115 e com o sensor do multímetro.	55
5.7	Resultados observados na utilização do sistema motor.	55
5.8	Comparação do valor medido com o termopar do tipo K conectado à placa ADS 1115 e com o sensor do multímetro.	56
5.9	Comparação do valor medido com o LM35 conectado à ADC do <i>Esp8266</i> e com o sensor do multímetro.	56
A.1	<i>Breadboard</i> ou placa de ensaio.	63
A.2	Circuito do sistema água montado em <i>breadboard</i>	64
A.3	Circuito do sistema motor montado em <i>breadboard</i>	64
A.4	Circuito do sistema ar montado em <i>breadboard</i>	65
B.1	Áreas principais da interface desenvolvida.	67
B.2	Apresentação do menu em ecrãs de dimensões reduzidas.	68
B.3	Área de trabalho relativa à opção “Temperaturas”.	68
B.4	Área de trabalho relativa à opção “Painel Solar”.	69
B.5	Área de trabalho relativa à opção “Histórico”.	69

Capítulo 1

Introdução

O presente capítulo pretende apresentar o âmbito de desenvolvimento desta dissertação, expor o porquê da sua realização e quais os objetivos que se pretendem alcançar com a mesma. Na parte final do capítulo descreve-se a estrutura do presente documento.

1.1 Enquadramento

Nos dias de hoje a tecnologia tem um papel fundamental no quotidiano dos cidadãos, sendo uma das principais propensoras de qualidade de vida dos mesmos. Neste ponto destaca-se o aumento de conforto e o elevado número de tarefas autónomas dentro das habitações, tendo a domótica contribuído amplamente para isso.

A palavra domótica deriva das palavras *Domus* (casa) e *Robótica* (estudo do uso e programação de robots), ou seja, a domótica define-se como a possibilidade de controlo de forma automática das habitações tornando-as no que vulgarmente se costuma designar por “casas inteligentes”.

De acordo com a definição explanada anterior pode-se prever que as *Smart Green Houses* são cada vez mais o futuro. Isto deve-se ao facto de assim se poderem automatizar muitas tarefas e tornar outras tantas autónomas. Deste modo a necessidade de implementação de sistemas de domótica nas habitações atuais torna-se uma necessidade real. Para colmatar essa lacuna existem no mercado várias soluções de domótica que podem ser implementadas em qualquer habitação e controlar as mais variadas gamas de produtos, desde máquinas de café até caldeiras e bombas de calor.

Analisando a realidade atual surge o tema de controlar e monitorizar vários sistemas dentro da habitação. Este assunto tem principal relevância no que toca ao controlo de custos e à otimização de recursos dentro da habitação. Tendo esta problemática em mente, e ao mesmo tempo pensando sempre no ambiente e nas emissões de gases de efeito de estufa, surge o desafio de o solucionar, encontrando a melhor solução possível para tal. Assim, pretende-se que exista um processamento de dados em tempo real para que seja possível a monitorização e o controlo de vários sistemas, tais como a escolha da opção de produção de água quente sanitária (AQS), o aquecimento do ar entre outros, tendo sempre em mente o menor impacto ambiental e ao mesmo tempo o menor custo possível.

1.2 Motivação

Tendo em conta a crescente preocupação com a proteção do meio ambiente e a diminuição dos gases de efeito de estufa, os órgãos de soberania dos vários países do mundo unem esforços e realizam tratados e protocolos com vista a diminuir a emissão destes gases e em proteger o meio ambiente, apelando ao uso de energias renováveis e ao consumo racional de energia. Apesar do carácter de grande relevância que estas medidas têm, a grande mudança passa por desenvolver mecanismos que possam ajudar os cidadãos a fazer um consumo mais racional e mais amigo do ambiente, isto mesmo sem que eles o notem.

Na União Europeia a emissão de gases de efeito de estufa, desde 1990, tem vindo a diminuir, passando de 5665,5 milhões de toneladas de CO_2 equivalente para 4282,1. No entanto Portugal não verifica a mesma tendência que a União Europeia, sendo que o número de 2014 (60,7 milhões de toneladas de CO_2) é ligeiramente superior ao de 1990 (64,6 milhões de toneladas de CO_2), no entanto o histórico revela que houve um aumento de 1990 até 2005, onde se registou um máximo (88,2 milhões de toneladas de CO_2), tendo depois existido uma redução significativa até 2014 [14].

Aliado a isso existe ainda o facto de na União Europeia, cerca de 25% da energia total consumida ser utilizada no sector habitacional. Em Portugal a percentagem é mais baixa, cerca de 16,5%, no entanto poderá ser ainda reduzida. De realçar que estes dados são referentes ao ano de 2014, último ano com estatísticas realizadas pela Comissão Europeia [14].

Em Portugal, de acordo com dados do ano de 2010, existe um consumo muito pouco significativo de eletricidade no aquecimento de águas e no aquecimento e arrefecimento do ar (cerca de 13%) quando comparados com outros sectores. Na figura 1.1 pode ser observado esse facto. De notar, ainda, o grande peso do sector da “Cozinha” e dos “Equipamentos Eléctricos”, facto que se deverá ao elevado número de eletrodomésticos já datados presentes nas habitações dos portugueses [1].

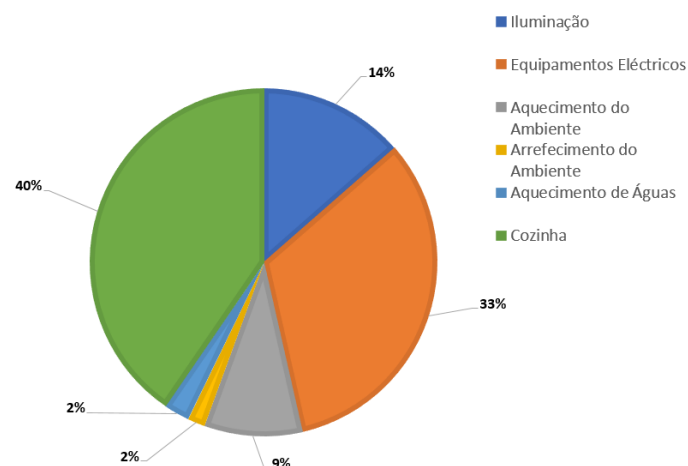


Figura 1.1: Distribuição do consumo doméstico de Eletricidade por tipo de utilização - Portugal, 2010 (Adaptado de [1])

1.3 Objetivos

Este trabalho tem como objetivo principal o desenvolvimento de um equipamento que seja possível de instalar em sistemas de aquecimento de água e de ar já existentes, para que faça o controlo da forma mais ecológica e económica possível. Este equipamento tem, ainda, de poder ser monitorizado e controlado via web.

Para a concretização deste objetivo foi definida uma metodologia que contém várias passos que devem ser seguidos de modo a facilitar o processo e a torná-lo mais organizado. Os passos a seguir são:

- Estudo das soluções existentes, de modo a averiguar as suas vantagens e desvantagens;
- Proposta de uma solução tendo em conta todos os casos analisados anteriormente;
- Desenvolvimento da solução proposta;
- Implementação da solução proposta;
- Realização de testes à solução proposta e implementada anteriormente;
- Análise dos resultados obtidos inerentes à realização dos testes.

1.4 Estrutura do documento

A presente dissertação além deste capítulo encontra-se organizada em mais cinco capítulos e dois apêndices:

- Capítulo 2 - Conceitos Fundamentais e Estado da Arte
No capítulo 2 apresentam-se os conceitos fundamentais para o bom entendimento do trabalho desenvolvido e também se apresentam algumas soluções já existentes. No final do capítulo é apresentada uma revisão a um artigo e a uma dissertação de outros autores;
- Capítulo 3 - Arquitetura do Sistema
No capítulo 3 apresenta-se a arquitetura concebida no sistema a implementar, assim como os intervenientes da mesma;
- Capítulo 4 - Implementação
No capítulo 4 apresentam-se e explicam-se todas as fases de implementação da solução proposta de modo a atingir todos os objetivos propostos. No início do capítulo faz-se também uma apresentação de todo o *hardware* e *software* utilizado ao longo da implementação do sistema;
- Capítulo 5 - Análise de Desempenho
No capítulo 5 apresentam-se os casos de estudo criados e a análise de desempenho aos mesmos;
- Capítulo 6 - Conclusão
No capítulo 6 é feita uma avaliação ao trabalho desenvolvido durante a dissertação tendo em conta o resultado final, apresenta-se ainda, sugestões de trabalho futuro e melhorias ao trabalho desenvolvido;

- Apêndice A - Circuitos do Sistema Proposto
No apêndice A são abordados os circuitos desenvolvidos ao longo da dissertação;
- Apêndice B - Interface *web*
No apêndice B é abordada a organização da interface assim como a sua programação.

Capítulo 2

Conceitos Fundamentais e Estado da Arte

Neste capítulo apresentam-se alguns conceitos fundamentais ao bom entendimento do trabalho desenvolvido. Deste modo, encontram-se representadas breves descrições dos principais conceitos de Fenómenos de Transferência de Calor, secção 2.1, e também alguma informação sobre algoritmos, secção 2.2.

Na secção 2.3 irão ser abordadas algumas soluções de domótica já existentes no mercado, dando-se especial ênfase ao X10 e ao KNX. Posteriormente, na secção 2.4, são apresentadas algumas soluções de aquecimento de águas sanitárias que existem no mercado.

No final do presente capítulo, nas secções 2.5 e 2.6, irão ser revistos trabalhos realizados por outros autores num âmbito idêntico ao da presente dissertação. No fim de cada revisão irá ser feita uma análise crítica às opções tomadas por esses autores.

2.1 Fenómenos de Transferência de Calor

Transferência de Calor é o fenómeno onde energia térmica transita devido a uma diferença de temperatura registada no meio ou entre meios, meios estes que podem ser sólidos ou líquidos [2].

Existem três modos de transferência de calor: Condução, Convecção e Radiação. Para o âmbito da presente dissertação apenas interessam os dois primeiros. De seguida apresenta-se uma breve descrição de cada um dos modos [2].

2.1.1 Condução

O fenómeno da condução de calor acontece sempre, e só, em meio estacionário quando se verifica um gradiente de temperatura nesse meio. O meio pode ser sólido ou líquido [2]. Na Figura 2.1 pode ser observado uma condução de calor através de um corpo sólido que tem uma face a uma temperatura diferente da outra.

2.1.2 Convecção

O fenómeno da convecção de calor acontece entre uma superfície e um fluido em movimento, acontece sempre que os dois estiverem a temperaturas diferentes [2]. Na Figura

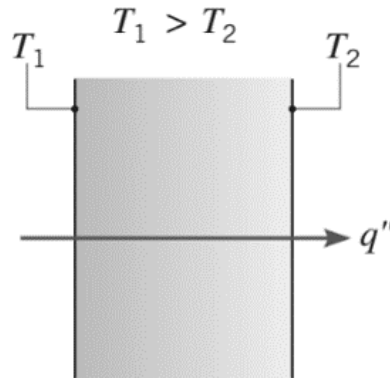


Figura 2.1: Condução de calor num sólido com duas faces a temperaturas diferentes (Adaptado de [2])

2.2 pode ser observado uma convecção de uma superfície para um fluido em movimento (não estacionário).

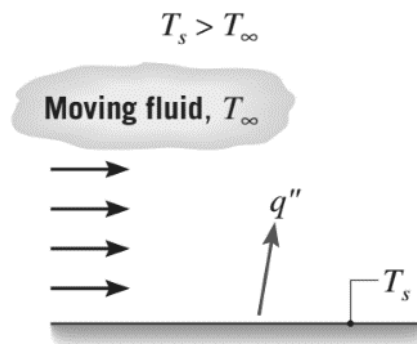


Figura 2.2: Convecção de uma superfície para um fluido em movimento (Adaptado de [2])

2.2 Algoritmos

Algoritmos de controlo são uma realidade cada vez mais presente e importante nos dias de hoje. Estes permitem-nos automatizar tarefas de modo a facilitar algum trabalho humano e até fazer tarefas de maior precisão que não são possíveis de ser feitas por pessoas.

Os algoritmos hoje em dia estão presentes numa panóplia de equipamentos que vão desde *robots* e equipamento de alta tecnologia até equipamentos mais simples e vulgares como o micro-ondas. De seguida irá ser apresentado um tipo de algoritmos.

2.2.1 Fuzzy Logic

Fuzzy Logic do Português Lógica Difusa, é um tipo de lógica que trabalha com verdades parciais, isto é, a lógica booleana trabalha com totalmente verdade ou totalmente falso (0 ou 1), na lógica difusa existem valores entre o totalmente verdadeiro e o totalmente falso (de 0 a 1). A Figura 2.3 mostra o dito anteriormente [15].

A lógica difusa é hoje em dia implementada nas mais diversas áreas dada a sua simplicidade e os reduzidos custos associados à sua implementação.

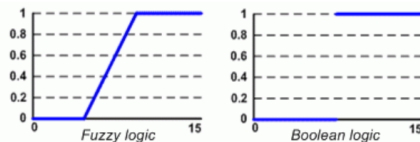


Figura 2.3: Diferenças entre lógica difusa e lógica booleana [3]

2.3 Principais soluções de domótica

Existem várias soluções de domótica no mercado, algumas abertas e outras proprietárias. No entanto de entre as mais utilizadas há duas que se destacam, são estas o KNX - Konnex e o X10.

O KNX - Konnex destaca-se devido à sua abrangência e integração, enquanto que o X10 destaca-se mais por uma razão histórica, dado ter sido uma solução pioneira.

2.3.1 KNX - Konnex

O protocolo KNX é uma junção de três outros protocolos mais antigos, EIB (*European Interconnection Bus*), EHS (*European Home Systems*) e BatiBus. Esta junção deveu-se à necessidade de um protocolo padrão nas instalações de domótica. Este protocolo pertencendo à categoria de tecnologia de acesso aberto, e não proprietário, garante a interação com produtos de diferentes fabricantes. Outras características importantes são o facto de suportar quatro meios de comunicação distintos e de ser um sistema modular e reconfigurável [16][17][18].

Os possíveis meios físicos de comunicação são:

- Cabo de par entrançado (*twisted pair*);
- Rede de energia elétrica (*Power Line*);
- Rádio Frequência;
- Ethernet.

2.3.2 X10

O protocolo X10 é, também, um protocolo de comunicação aberto e normalizado, o que garante interação com produtos de diferentes fabricantes. O meio utilizado para realizar a comunicação é a rede de distribuição de energia. Os módulos existentes são conectados

à rede elétrica e recebem sinais de rádio frequência de um comando. Este sinal propaga-se através desta até comunicar com o aparelho de destino e transmitir a ordem pedida. No entanto o sinal ao ser enviado torna-se acessível a todos os dispositivos conectados, tal como a rede elétrica está acessível em toda a habitação [19].

Para contornar o problema do endereçamento do sinal, foi criado um sistema de códigos com dois registos. O primeiro registo são letras maiúsculas de “A” a “P”, e diz respeito ao código de casa. O segundo registo é o código de aparelho e é representado por algarismos de 1 a 16. Deste modo, conjugando todas as combinações possíveis destes dois registos é possível obter duzentos e cinquenta e seis aparelhos conectados à mesma rede elétrica com endereços diferentes.[19]

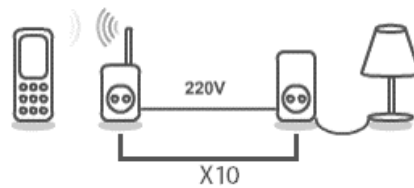


Figura 2.4: Esquema de comunicação do protocolo X10 (Adaptado de [4])

2.4 Sistemas de Águas Quentes Sanitárias existentes

Atualmente existem as mais variadas soluções para produzir água quente tanto no âmbito habitacional como empresarial. Estas soluções podem ser desde caldeiras, termoacumuladores elétricos, esquentadores, painéis solares e bombas de calor.

Com estes equipamentos é possível criar vários sistemas combinados de aquecimento de águas sanitárias; vários já existem no mercado. De seguida apresenta-se uma breve lista das combinações, resultantes em sistemas de AQS, mais usuais.

- **Caldeira, Termoacumulador elétrico, Esquentador**

O princípio de funcionamento difere do esquentador e da caldeira para o cilindro. Enquanto que o esquentador (Figura 2.5) aquece um fluído em movimento numa serpentina, isto é, há um caudal de água fria que entra no esquentador e percorre uma serpentina, por baixo desta existe um combustível (normalmente gás) que está a ser queimado produzindo calor, este é passado por condução para o tubo e posteriormente por convecção para o fluído em movimento no interior da serpentina.

No caso dos termoacumuladores elétricos (Figura 2.6) o princípio de funcionamento é um pouco diferente, existindo um termoacumulador onde a água vai ser aquecida toda de uma vez para posteriormente ser usada. O princípio de funcionamento é uma serpentina que possui uma resistência elétrica no seu interior e que está no interior do termoacumulador fazendo aquecer a água que está em contacto com esta, o que origina correntes de convecção dentro do depósito. A serpentina aquece com recurso a energia elétrica.

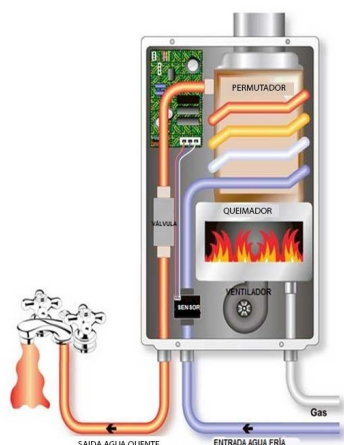


Figura 2.5: Esquema de funcionamento de um esquentador. (Adaptado de [5])

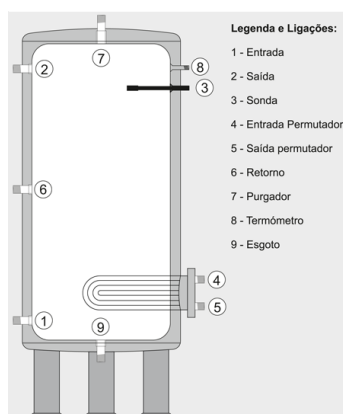


Figura 2.6: Esquema de um termoacumulador elétrico. (Adaptado de [6])

Existe ainda uma pequena diferença entre a Caldeira e o Esquentador, esta prende-se com o facto de uma caldeira, hoje em dia, possuir uma válvula de três vias que faz o aquecimento de água para águas sanitárias e para o aquecimento central, por norma dando sempre prioridade às águas sanitárias. A antiga caldeira que tinha um termoacumulador tornou-se redundante nos dias de hoje dado a crescente implementação de sistemas térmicos solares.

• Solar Térmico

Esta tecnologia começa a ser muito usual especialmente combinada com esquentadores ou com cilindros, formando um sistema mais amigo do ambiente e com recurso a muito menos combustível, dado que parte, ou até mesmo todo o aquecimento é feito através da energia solar. Existem duas tecnologias de aquecimento de águas sanitárias através de energia solar que são: Termossifão e Circulação Forçada.

A tecnologia de Termossifão (Figura 2.7) consiste num termo-acumulador colocado ao lado do painel solar onde entra água sanitária fria e sai água quente. Dentro deste termoacumulador existe uma serpentina que contém um líquido refrigerante, que aquece através da radiação que incide sobre o painel solar. O líquido quente

transmite, depois, a sua energia à água que passa no termoacumulador, através de convecção. Esta tecnologia é muito utilizada em locais onde o espaço é reduzido ou não exista espaço para instalar um termoacumulador dentro da habitação.

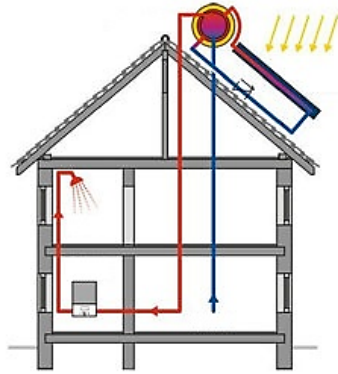


Figura 2.7: Esquema de funcionamento de um sistema solar com tecnologia de Termossifão. (Adaptado de [7])

A tecnologia de Circulação Forçada (Figura 2.8) é em tudo idêntica à tecnologia Termossifão diferindo em alguns pontos, tais como o termoacumulador onde se dá a transferência de energia do circuito solar térmico para a água sanitária estar, por norma, localizado dentro da habitação ao invés do exterior.

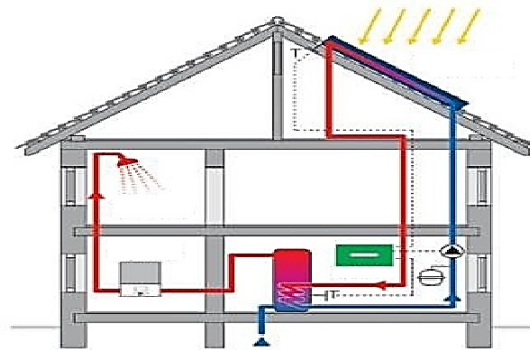


Figura 2.8: Esquema de funcionamento de um sistema solar com tecnologia de Circulação Forçada. (Adaptado de [7])

De realçar que sem exposição solar torna-se mais difícil realizar o aquecimento da água, então qualquer uma das tecnologias acima referida possui um sensor que deteta a temperatura da água no termoacumulador e a faz reaquecer numa caldeira ou esquentador, até à temperatura previamente programada, antes de chegar ao ponto onde é solicitada pelo utilizador.

De notar que a opção por instalar esta tecnologia nas habitações tem vindo a

aumentar nos últimos anos.

- **Bomba de Calor**

Esta tecnologia é das mais recentes, fazendo uso do princípio do ar condicionado para aquecer a água, como é possível observar na Figura 2.9. Usando este princípio a bomba de calor vai trocar o calor com a água sanitária através de uma serpentina num termo-acumulador. O funcionamento do ar condicionado consiste num fluido refrigerante dentro de um circuito fechado, que começa por absorver o calor do ar, depois é comprimido de modo a aumentar o seu potencial térmico, posteriormente faz a troca de calor num permutador para a água, neste caso num termoacumulador, e finalmente como o fluido refrigerante perdeu a sua capacidade térmica passa por uma válvula de expansão, de modo a ficar totalmente arrefecido e assim estar preparado para receber novamente o calor do ar exterior. [20]

De notar a capacidade de incorporar esta tecnologia com a Solar Térmica de modo a obter um aproveitamento maior de energias renováveis.

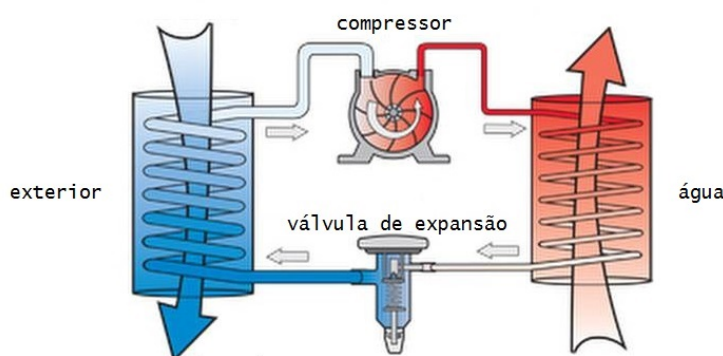


Figura 2.9: Esquema que ilustra o princípio de funcionamento de uma bomba de calor. (Adaptado de [8])

É possível combinar os sistemas acima mencionados, de modo a conseguir aquecer a água com recurso a menos fontes de energia fóssil e a mais fontes renováveis, poupando o ambiente e tendo um custo monetário bastante inferior, dado que as fontes de energias renováveis estão disponíveis de forma gratuita para todos os habitantes do planeta Terra.

Tendo em conta que os sistemas que utilizam fontes de energia renovável são bastantes complexos, é conveniente conseguir definir *set points* e fazer monitorização de maneira generalizada sem descentralizar o controlo por vários equipamentos. No mercado já existem algumas soluções que fazem uso de sensores de vários tipos para fazer o controlo automático de aquecimento, ou reaquecimento da água. No entanto não existe uma solução que conjugue tudo num único sítio e que possa ser monitorizado e controlado pelo utilizador.

Do que já existe no mercado destacam-se sistemas integradores que conjugam vários dos sistemas referidos anteriormente. Na figura 2.11 pode ser visto um exemplo de um sistema desses. Este sistema possui um termoacumulador que contém 2 serpentinas de aquecimento de água, uma proveniente de um sistema solar térmico e outra proveniente de um recuperador de calor, e um esquentador na saída da água do termo-acumulador para o caso de esta não ter temperatura suficiente para satisfazer o pedido do utilizador.

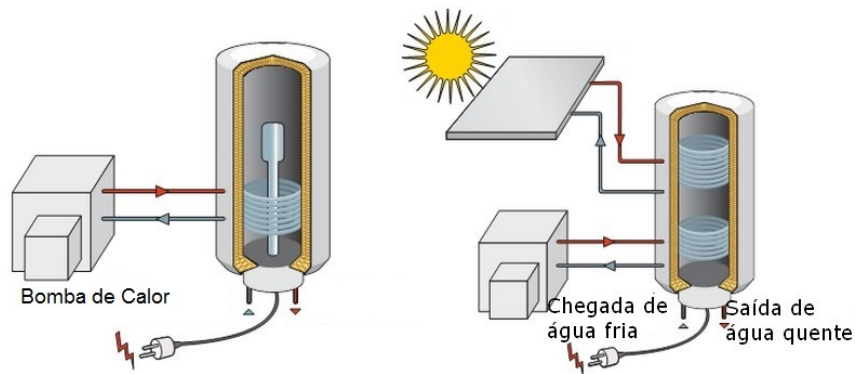


Figura 2.10: Esquema de sistemas com recurso a mais do que uma fonte de energia para aquecimento de águas sanitárias. (Adaptado de [9])

É possível observar que existem vários sensores de temperatura ao longo do circuito, uma unidade de controlo e duas bombas de recirculação de fluido refrigerante. A unidade solar térmica possui uma unidade de controlo independente do resto do sistema, isto é, mede a temperatura a que fluido refrigerante está, mede a temperatura a que a água dentro do termo-acumulador está e com base nesses dados liga, ou não, a bomba para fazer a recirculação do fluido refrigerante nesse circuito. O recuperador de calor tem o mesmo sistema de funcionamento, mas de controlo independente do sistema solar térmico. À saída do termo-acumulador existe ainda outro sensor de temperatura que vai fazer acionar o esquentador se a água não estiver à temperatura pretendida pelo utilizador.

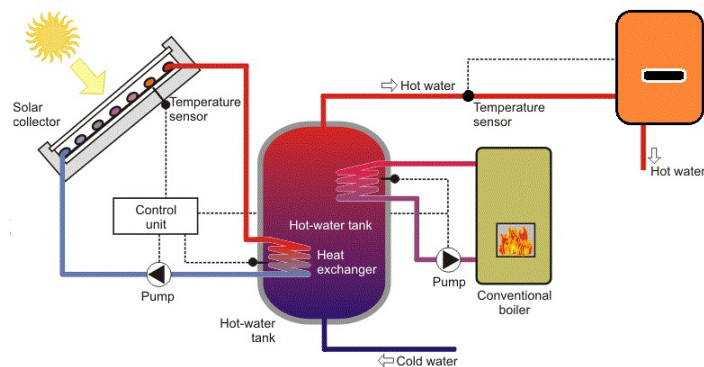


Figura 2.11: Esquema de um sistema que recorre a múltiplas fontes de energia para o aquecimento de águas sanitárias. (Adaptado de [10])

2.5 *Controlling the average residential electric water heater power demand using fuzzy logic*

Este artigo, datado de 1999 e da autoria de B.J. LaMeres, M.H. Nehrir & V. Gerez, foi desenvolvido no âmbito da lógica difusa aplicada ao controlo do aquecimento de água através do uso da eletricidade. A motivação para a realização deste artigo foi a alta

dependência de eletricidade em certos períodos do dia, dadas as rotinas normais de uma população empregada, o que faria com que as companhias distribuidoras de eletricidade teriam de ter disponível uma potência muito maior para fornecer às casas, devido ao elevado uso da mesma em certos períodos do dia. Deste modo cria-se um desperdício de recursos enorme, logo faria todo o sentido criar um controlo inteligente capaz de utilizar a energia ao longo do dia para manter a água sempre quente e evitar usar energia nos períodos, que já se sabem, ter maior utilização.

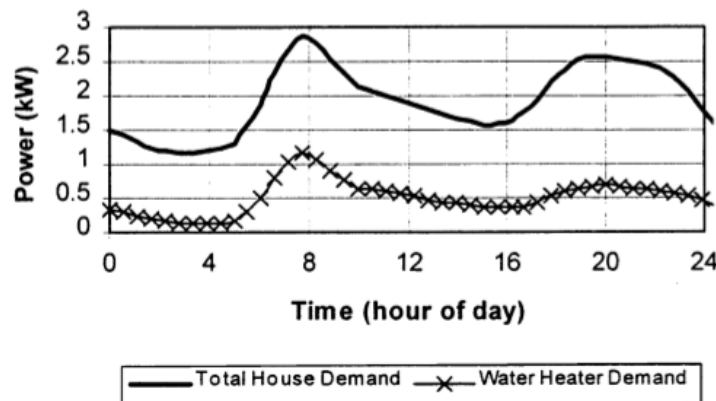


Figura 2.12: Potência despendida no aquecimento de águas sanitárias e no total em função da hora ao longo do dia [11]

Para solucionar o problema apresentado os autores decidiram utilizar um algoritmo de lógica difusa, tendo como parâmetros de entrada o horário de maior uso de energia elétrica, a temperatura da água, a temperatura de conforto e a temperatura máxima. Tendo em conta estes parâmetros foram criadas 22 regras que fazem o controlo da saída, que é a potência fornecida à resistência elétrica que aquece a água. O diagrama de blocos presente na figura 2.13 é explicativo do dito anteriormente, mostrando ainda que dois dos sinais de *input* não são de lógica booleana e estão divididos em vários graus de verdade, isto é já foi realizada a “fuzificação” das entradas.

Aplicando as regras referidas anteriormente é possível fazer uma otimização da energia consumida para aquecer a água e também gastar de forma mais controlada os recursos disponibilizados pela empresa distribuidora de energia.

Um exemplo de uma das regras usadas foi:

If (Demand is low) and (Water-Temp is h-warm) then (Power is avg) . Isto é, se existir pouca procura de eletricidade para aquecer a água, isto é, fora das horas de uso intensivo, e a temperatura da água já está quente mas não na temperatura máxima, então a potência dada ao elemento resistivo que vai aquecer a água, é média. Tendo em conta estas regras e recorrendo a simulações, em ambiente real, obteve-se o gráfico presente na Figura 2.14. Neste gráfico é possível observar várias diferenças de consumo de eletricidade ao longo do dia, em comparação com a Figura 2.12. A principal diferença reside nas horas de maior utilização de eletricidade o uso ser reduzido, distribuindo assim o consumo ao longo do dia.

Foi ainda realizada outra simulação considerando dois casos. O primeiro caso em

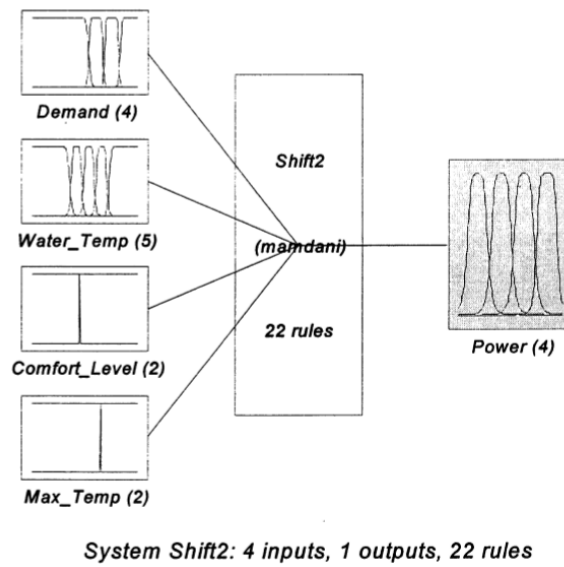


Figura 2.13: Diagrama de blocos do controlador implementado com *Fuzzy Logic* [11].

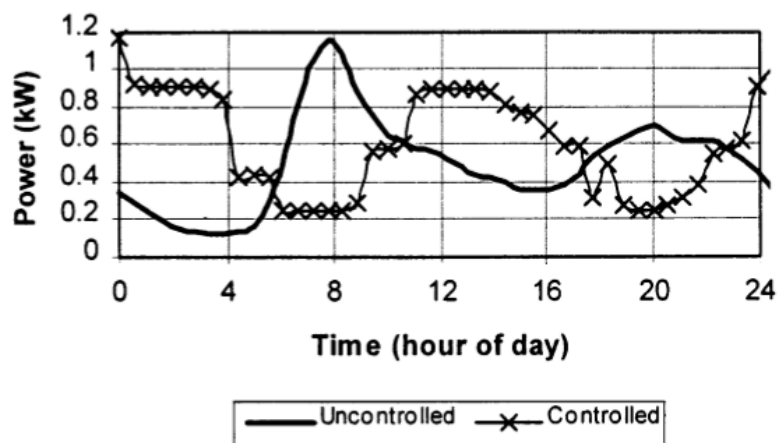


Figura 2.14: Comparação de um equipamento controlado e um não controlado [11]

que existe um equipamento controlado e outro não controlado e um segundo caso em que existem dois equipamentos não controlados. O resultado desta simulação pode ser visto na Figura 2.15. Da interpretação desta observa-se que deixam de existir picos de consumo de eletricidade e a curva fica mais suave ao longo do dia.

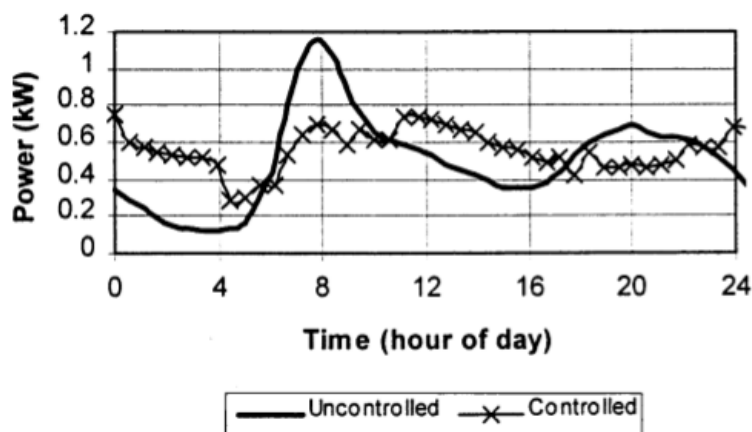


Figura 2.15: Comparação entre dois equipamentos controlados e dois equipamentos em que um é controlado e o outro não [11].

Em suma os autores concluem que é de facto possível harmonizar o consumo de eletricidade ao longo do dia e não criar grandes picos de consumo, no entanto o balanço final de energia despendida pode resultar num maior gasto energético.

2.6 *Development of Automatism and intelligent control algorithms, for monitoring and control of housing components*

Esta dissertação apresentada à Universidade de Aveiro datada de 2016 e da autoria de André Filipe da Silva Pires foi realizada no âmbito do desenvolvimento de algoritmos para controlo inteligente da temperatura dentro de habitações. A motivação para a realização desta dissertação prende-se com o ambiente e quantidade de energia usada no sector habitacional, que apesar de ser uma pequena parcela da gasta em todos os sectores, pode ser bastante reduzida devido a sistemas com alguma inteligência e automatizados, isto é não dependendo de ação do homem.

Para solucionar o problema encontrado e descrito anteriormente, o autor propõe a implementação de um algoritmo que a partir da aquisição de vários dados de entrada consegue atuar e controlar os sistemas de aquecimento de ar da habitação. Os parâmetros de entrada são:

- Temperatura exterior;
- Temperatura interior;
- Temperatura geotérmica;

- Temperatura dos circuitos de água para aquecimento;
- Caudal de água;
- Caudal de ar;
- Nível de CO_2 .

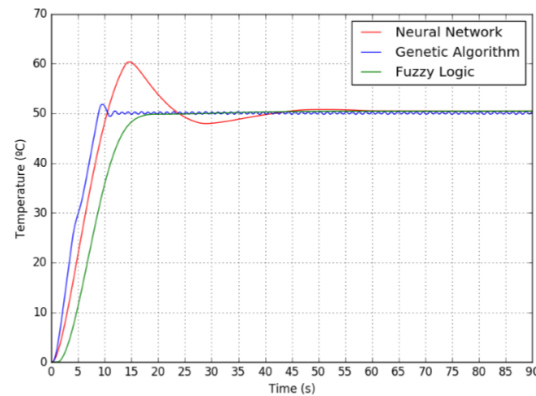


Figura 2.16: Comparação da implementação dos vários algoritmos [3].

O autor desde o início decidiu usar um controlador PID e um algoritmo capaz de controlar dinamicamente o valor dos coeficientes associados a esse controlador. Para melhor escolher o algoritmo foram realizados testes a três tipos de algoritmos; Algoritmos *Fuzzy Logic* ou de lógica difusa, algoritmos genéticos e algoritmos de redes neurais. Os resultados dos testes feitos aos algoritmos podem ser vistos na Figura 2.16 e na Tabela 2.1.

Tabela 2.1: Valores obtidos pelo autor nos vários testes que realizou.

Algoritmo	<i>Overshoot (%)</i>	<i>Rise Time (s)</i>	<i>Steady-State Error (%)</i>
<i>Fuzzy Logic</i>	$\simeq 0$	$\simeq 17$	$\simeq 0$
Genético	3,03	$\simeq 9$	0,28
Redes Neurais	16,36	$\simeq 14$	$\simeq 0$

Da análise dos dados obtidos o autor pôde concluir, que o algoritmo *Fuzzy Logic* seria o mais indicado devido a não ter *overshoot* e ao erro em regime estacionário ser praticamente nulo, mas o facto de necessitar de inúmeras regras para o bom funcionamento compromete a viabilidade do mesmo; o algoritmo genético foi logo posto de parte devido ao seu comportamento em regime estacionário não ser constante; o algoritmo de redes neurais também possui um erro em regime estacionário praticamente nulo e cada vez mais próximo de zero, também devido a ser considerado o histórico de valores, o *rise time* é mais rápido que o do algoritmo de *Fuzzy Logic*.

Tendo tomado em conta os dados acima apresentados, o autor decidiu usar um algoritmo de redes neurais excluindo o de *Fuzzy Logic* devido às inúmeras regras que seriam precisas para controlar o mesmo, e excluindo o algoritmo genético devido ao comportamento não linear em regime estacionário. Para programar o algoritmo o autor decidiu usar a linguagem de programação PYTHON.

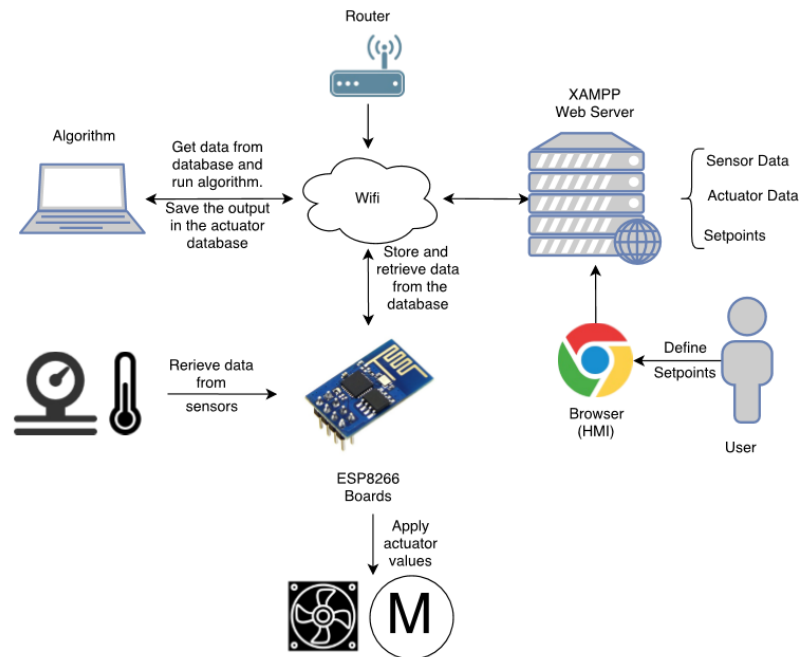


Figura 2.17: Representação da arquitetura do modelo construído pelo autor [3].

Para implementar a solução o autor seguiu a arquitetura presente na Figura 2.17. De notar na mesma figura o uso de uma base de dados, criada pelo autor através do programa XAMPP, e uma página HTML também criada pelo autor para definir os *set points* para o sistema e para fazer a monitorização do mesmo.

O sistema desenvolvido foi posteriormente testado com recurso a um modelo de escala construído pelo autor. As interações entre os equipamentos são demonstradas pela figura 2.18, onde se pode observar os três diferentes objetos, o computador do utilizador que tem o *browser*, o *web server* e os ESP's necessários que fazem o controlo e adquirem os dados necessários.

Concluindo o autor cumpriu todos os objetivos a que se propôs e salienta que para trabalho futuro poderiam ser desenvolvidos todos os algoritmos tidos como hipótese, para verificar se realmente haveria diferenças de desempenho e comportamento dos mesmos em casos reais.

Tendo em conta tudo o que foi analisado anteriormente é possível concluir que já existem muitas soluções no mercado. Deste modo a presente dissertação propõem-se a criar uma solução que consiga combinar todos os sistemas acima referidos de forma integradora. A solução também será capaz de, no futuro se necessário, contemplar algoritmos inteligentes para um modo de funcionamento autónomo.

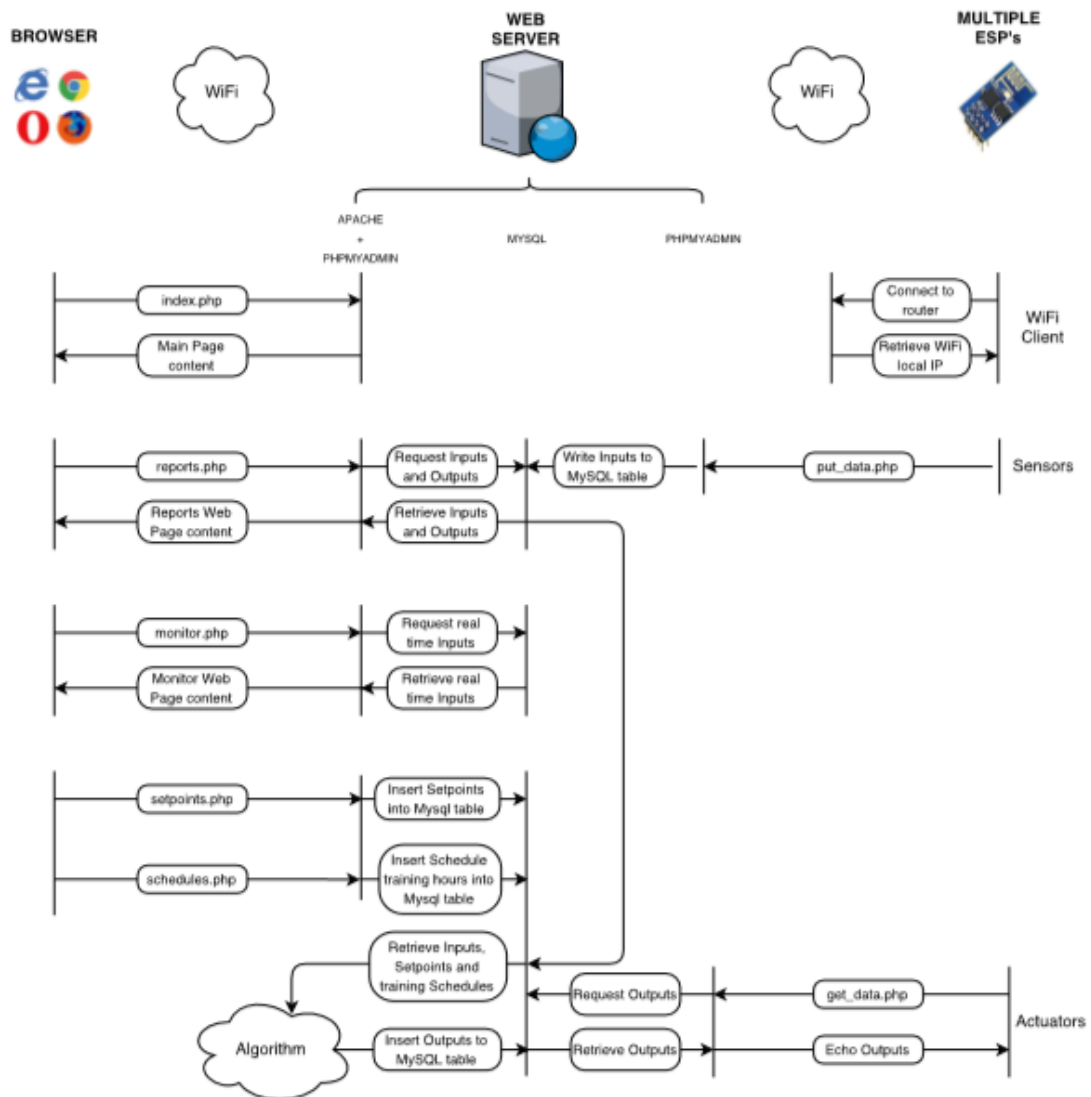


Figura 2.18: Esquema da solução implementada pelo autor [3].

Capítulo 3

Arquitetura da Solução Proposta

Neste capítulo irá ser apresentada a solução proposta, secção 3.1, assim como os requisitos da mesma. Será também apresentada a arquitetura proposta para a solução, sendo esta baseada no modelo Cliente/Servidor. Posteriormente serão mostradas as diversas funcionalidades do Cliente e do Servidor, assim como, na secção 3.2 as interações entre estes.

3.1 Solução proposta

A solução proposta é baseada numa habitação genérica, como é possível observar na Figura 3.1. Deste modo, é proposto monitorizar e controlar grandezas a partir de uma interface.

Algumas das grandezas a monitorizar são:

- Valor da temperatura da água dentro do termoacumulador;
- Valor da temperatura dentro do circuito solar térmico;
- Valor da temperatura dentro do circuito da caldeira;
- Valor da temperatura dentro da habitação;
- Valor da temperatura fora da habitação.

Também é proposto atuar mecanismos na habitação. São eles:

- Aquecer a água dentro do termoacumulador, através de uma resistência elétrica;
- Fazer recircular, ou não, a água nos circuitos independentes, caldeira e solar térmico;
- Aquecer o ar da habitação;
- Fazer recirculação do ar dentro da habitação;
- Introdução de ar novo na habitação;
- Fazer rodar o painel solar.

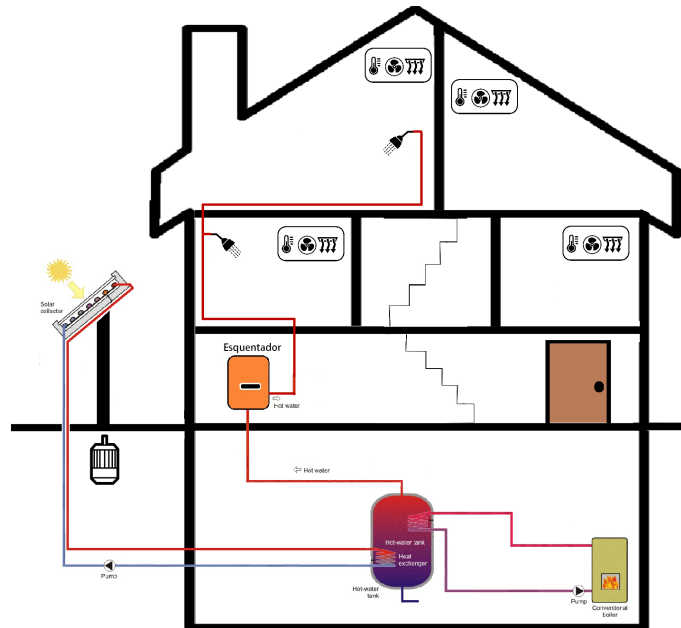


Figura 3.1: Habitação genérica com os vários sistemas prontos a monitorizar e controlar.

A solução proposta contempla ainda o desenvolvimento de uma interface para realizar o controlo e a monitorização. Para tal propõem-se que esta seja uma interface *web* e que a partir desta seja possível fazer todos os pontos acima mencionados, tanto em relação à monitorização como em relação ao controlo.

3.1.1 Arquitetura da solução proposta

A arquitetura proposta para a solução desenvolvida é baseada no modelo Cliente/Servidor. Este modelo é amplamente utilizado nos dias de hoje, sendo que muitos protocolos de comunicação o utilizam para comunicar, alguns exemplos são os casos do *File Transfer Protocol* (FTP), do *Simple Mail Transfer Protocol* (SMTP) e o *Hypertext Transfer Protocol* (HTTP) [21].

Deste modo, a escolha do modelo de comunicação para a arquitetura proposta, não implica um estudo adicional de qual o modelo de comunicação a usar, uma vez que o modelo Cliente/Servidor é um modelo já com provas dadas e bastante utilizado, tal como visto anteriormente.

Neste modelo possível existir vários clientes do mesmo servidor, o que vai de encontro às necessidades da solução proposta. Na Figura 3.2 pode ser vista a arquitetura da solução proposta, em que vários clientes fazem pedidos ao servidor e este responde a esses vários pedidos. O servidor também comunica com a Base de Dados que está presente nele. De notar que apenas existe um servidor podendo haver vários clientes, estando todos os intervenientes, clientes e servidores, ligados à mesma rede local.

Sendo o esquema genérico da arquitetura da solução proposta o apresentado na Figura 3.2 é pertinente enquadrar esse mesmo esquema agora num conceito real, para isso foi utilizada a Figura 2.11 que está na secção 2.4 representa um esquema de um sistema típico de uma instalação de AQS que recorre a múltiplas fontes de calor. De seguida

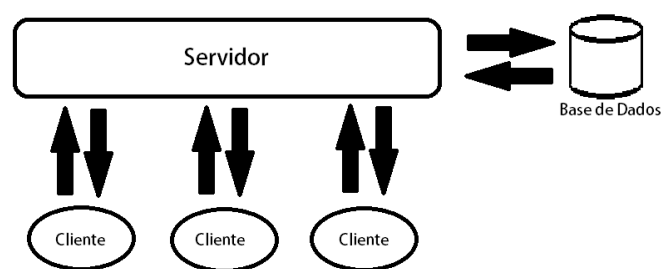


Figura 3.2: Modelo Cliente/Servidor aplicado à solução proposta.

é mostrado o mesmo esquema mas agora com os vários clientes, alguns instalados em certos locais, de modo a controlar aparelhos e a monitorizar grandezas físicas e outros a aceder ao servidor e a monitorizar e controlar as grandezas disponíveis. Também irá estar presente o servidor.

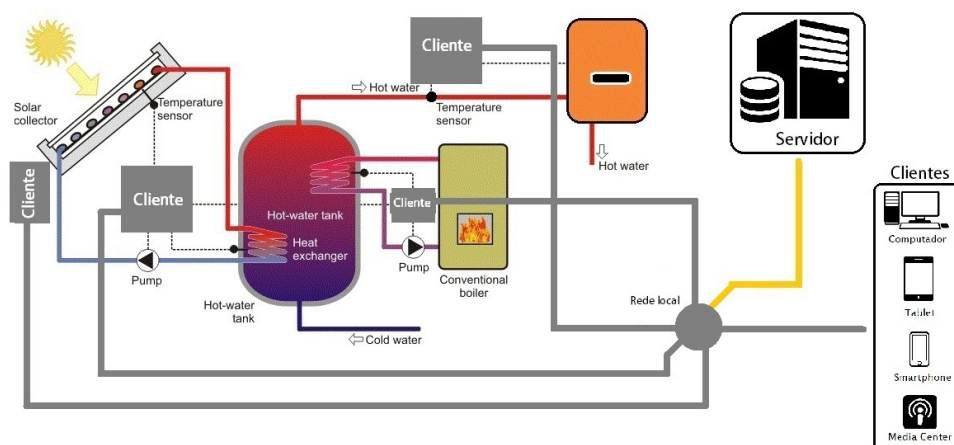


Figura 3.3: Arquitetura da solução proposta.

Na Figura 3.3 podem ser observados vários clientes. As ligações de clientes à rede local representam-se por linhas cinzentas, enquanto que a ligação do servidor representa-se por uma linha amarela. Há ainda que destacar o facto de existirem dois tipos de clientes diferentes, as diferenças irão ser explicadas na Secção 3.1.3.

3.1.2 Servidor

Este elemento do sistema, designado de Servidor, é constituído por dois outros servidores, um servidor base de dados e um servidor *web*. O servidor base de dados tem como propósito armazenar os valores atuais dos sensores e do estado dos equipamentos a controlar, o servidor *web* tem como propósito disponibilizar uma página *web* onde se possa monitorizar e controlar todo o sistema. Para realizar o armazenamento do estado das entradas (sensores) e saídas (equipamentos a controlar) existe uma base de dados. Na Figura 3.4 pode ser visto todos os constituintes do Servidor.

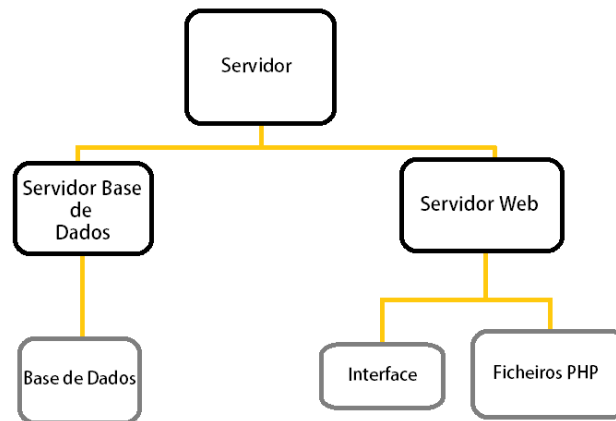


Figura 3.4: Elementos constituintes do servidor.

Conforme referido anteriormente, o propósito do servidor base de dados e servidor *web* é armazenar a base de dados e a interface *web* para qualquer utilizador poder controlar e monitorizar o sistema. Para tal acontecer é necessário alguns ficheiros PHP (*Hypertext Preprocessor*). Estes ficheiros auxiliares têm a função de escrever e ler da base de dados. Isto é, para na interface *web* estar disponível para ser monitorizado o valor de um sensor é preciso aceder a uma tabela na base de dados e ler o valor associado a esse sensor, assim como para controlar uma saída é necessário que se escreve na base de dados no campo destinado a essa saída. Os ficheiros PHP agilizam esse processo enviando as *query's* certas diretamente para o servidor base de dados, devido a funções já presentes na linguagem. Deste modo torna-se mais fácil aos clientes que possuem sensores enviar os seus valores para a base de dados e ler os valores das saídas a ativar.

A seguir, para resumir o explicado anteriormente, são apresentados os constituintes do servidor assim como os seus propósitos para o sistema.

- **Base de Dados** - Armazenar o estado atual de todas as entradas e saídas nos vários clientes;
- **Interface Web** - Possibilitar ao utilizador controlar e monitorizar todas as entradas e saídas dos vários clientes;
- **Ficheiros PHP** - Ficheiros auxiliares que agilizam a comunicação entre o servidor base de dados e o servidor *web*.

3.1.3 Cliente

Este elemento do sistema, designado de Cliente, pode ser separado em duas categorias diferentes. A primeira é o cliente que acede à interface *web* através do seu computador e consegue monitorizar, por exemplo, a temperatura a que a água está dentro do termo-acumulador ou controlar o aquecimento da mesma; a segunda é o **cliente controlador** que vai escrever e ler da base de dados, e com base nessas leituras vai atuar as saídas, por exemplo fazer comutar um relé para que a resistência elétrica possa funcionar e aquecer a água.

De seguida irá ser explicado todos os constituintes de cada categoria de cliente e as suas funcionalidades. De notar que a primeira categoria de cliente vai ser chamada de **cliente externo**.

Cliente externo

Esta categoria de cliente apenas acede à interface *web* para monitorizar e controlar o sistema, é portanto considerado o utilizador do sistema. A interface é disponibilizada através de um link que qualquer aparelho com um *browser* pode aceder, desde que esteja conectado à mesma rede LAN (*Local Area Network*) que o Servidor, isto é, pode ser acedido tanto através de um computador fixo como de um computador portátil, um telemóvel ou até mesmo um *Media Center* instalado na habitação do utilizador.

Na Figura 3.5 é possível ver o único constituinte desta categoria de cliente, o utilizador do sistema.



Figura 3.5: Elementos que constituem o **cliente externo**.

Cliente Controlador

Esta segunda categoria representa o cliente que tem periféricos conectados às suas entradas e saídas, e que vai fazer a comunicação com o servidor para atualizar na base de dados os estados destas. Para fazer a atualização recorre aos ficheiros PHP auxiliares já antes referidos. Conforme a resposta obtida pelos ficheiros PHP as saídas vão ser atuadas, isto é, comutar um relé para ligar a recirculação do ar, por exemplo. Na Figura 3.6 é possível observar os elementos que constituem esta categoria de cliente, de notar que as ligações a amarelo são físicas.

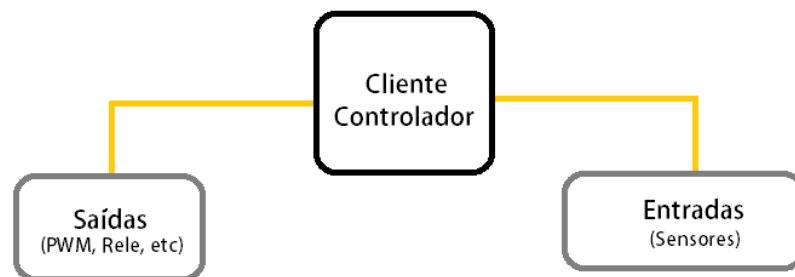


Figura 3.6: Elementos que constituem o **cliente controlador**.

Os elementos que constituem os Clientes controladores são as entradas e as saídas, como é ilustrado na Figura 3.6. As entradas são usadas para receber sinais de sensores de temperatura, de pressão, indutivos, entre outros. As saídas são usadas para enviar sinais a periféricos, isto é, *PWM (Pulse Wide Modulation)*, comutar relés entre outros.

Todos estes elementos têm o seu estado atual escrito na base de dados, devido aos pedidos enviados do cliente controlador para o servidor.

3.2 Interações na Solução Proposta

O funcionamento do sistema proposto tem por base o modelo cliente/servidor, tal como dito na secção 3.1.1, e os constituintes do sistema são os referidos em 3.1.2 e 3.1.3. Deste modo a presente secção irá explicar o funcionamento do sistema assim como as principais características do mesmo em relação à comunicação cliente/servidor.

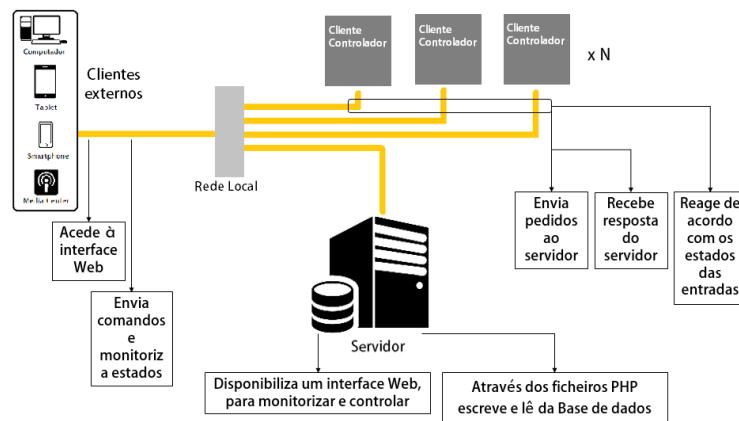


Figura 3.7: Interações entre os intervenientes da solução proposta.

Nesta solução o servidor encontra-se sempre à escuta e sempre que recebe um pedido executa-o e devolve resposta, se houver resposta a devolver. Deste modo o Servidor está sempre à espera de pedidos, tanto de clientes externos, como de clientes controladores. Na Figura 3.7 é possível agrupar as principais funções de todos os elementos da solução proposta.

Para um **cliente externo** aceder à página de controlo e monitorização terá de aceder a um ficheiro HTML que está disponível através do IP (*Internet Protocol*) do servidor seguido do diretório e nome do ficheiro. Uma vez tendo acedido à interface *web* o utilizador pode agora controlar e monitorizar todos os outros clientes controladores. Isto é possível porque a interface escreve e lê da base de dados em tempo real. Na Figura 3.8 é mostrado o anteriormente dito.

Deste modo o observado na Figura 3.8 pode ser resumido ao envio de várias mensagens, previamente formatadas, com os vários campos pedidos pela interface *web* ao servidor *web*. Assim o cliente externo irá poder monitorizar e controlar todos os estados. De notar que os pedidos que chegam ao servidor *web* são depois reencaminhados, por este, para o servidor base de dados. Estas interações podem ser vistas na Figura 3.9.

Assim como a partir da interface *web* é possível escrever e ler da base, então é possível controlar e receber valores dos **clientes controladores**. Estes estão constantemente

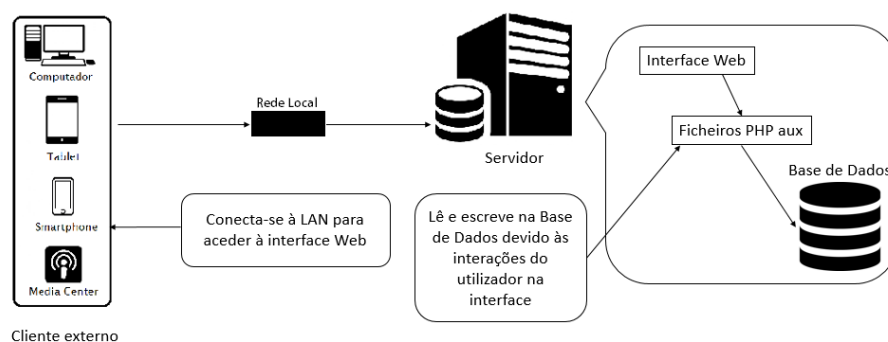


Figura 3.8: Interação do cliente externo com o servidor.

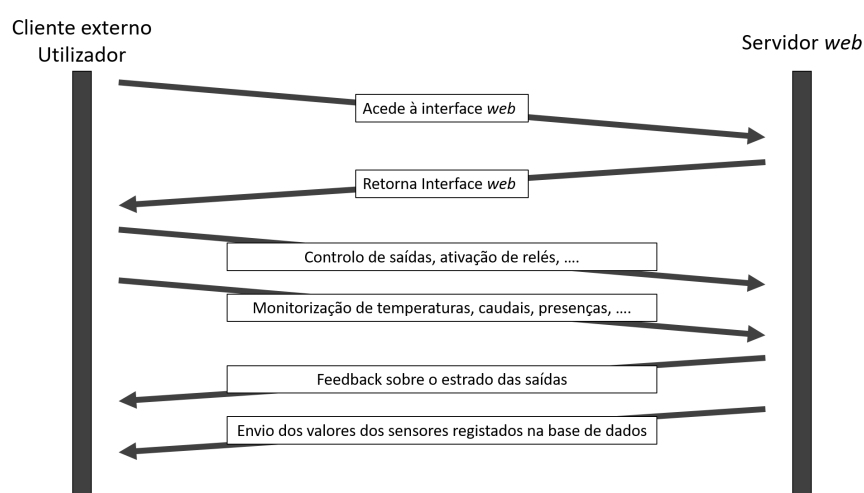


Figura 3.9: Mensagens trocadas entre o cliente externo e o servidor.

a enviar pedidos ao servidor e a receber respostas do mesmo, deste modo é possível monitorizar todos os sensores e controlar todas as saídas do sistema. Na Figura 3.10 é possível observar o dito anteriormente.

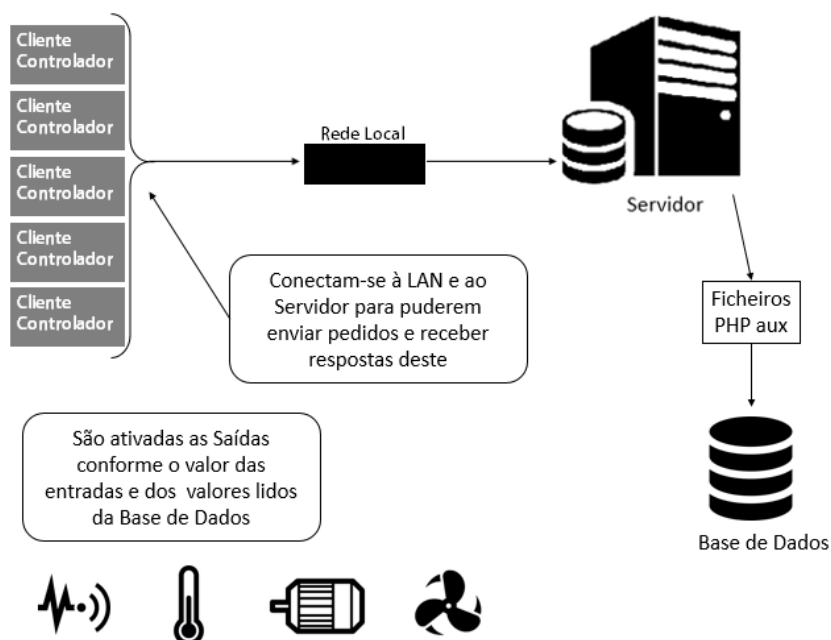


Figura 3.10: Interação do cliente controlador com o servidor.

Tal como são mostradas as mensagens trocadas entre cliente externo, utilizador, e servidor *web* faz também sentido mostrar as mensagens que são trocadas entre o cliente controlador e o servidor *web*. Estas mensagens podem ser vistas na Figura 3.11.

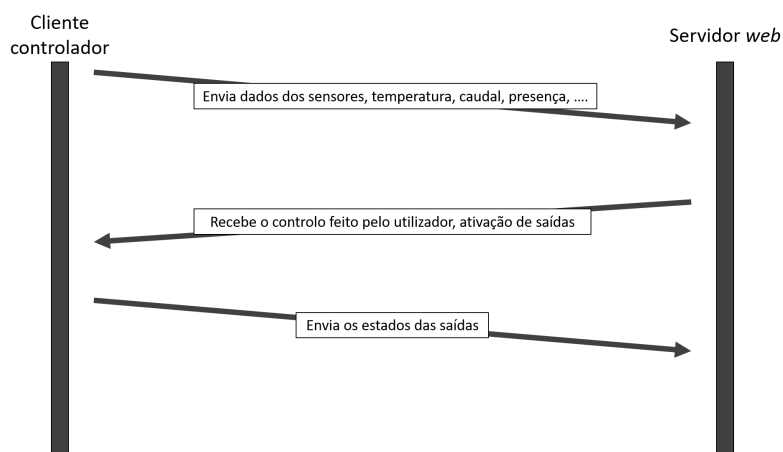


Figura 3.11: Mensagens trocadas entre o cliente controlador e o servidor.

O elemento servidor *web*, que faz a funcionar todo o sistema, tem de ser um servidor Apache. Deste modo todos os cliente podem comunicar com ele e este pode comunicar com todos. Assim todos os pedidos de que chegam ao servidor *web* que sejam pedidos

de atualização de valores na base de dados são reencaminhados para o servidor base de dados. Tal como todos os valores lidos da base de dados são pedidos efetuados ao servidor *web* que reencaminha os mesmos pedidos para o servidor base de dados. Deste modo todos os pedidos que chegam ao servidor *web* feitos pelos clientes controladores ou feitos pelos clientes externos, utilizadores, e sejam relativos à base de dados são reencaminhados para o servidor base de dados através do servidor *web*. Um esquema simplista de troca de mensagens entre o servidor *web* e o servidor base de dados está presente na Figura 3.12.

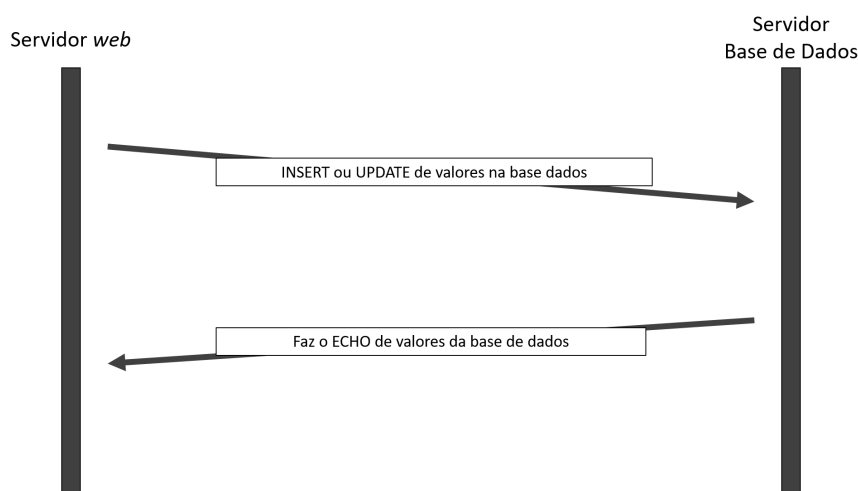


Figura 3.12: Mensagens trocadas entre o Servidor *web* e o Servidor Base de Dados

Concluindo, os **clientes controladores** recolhem e processam os dados dos sensores aos quais estão conectados, posteriormente enviam um pedido ao servidor *web* para atualizar na base de dados os dados por si recolhidos e tratados. O servidor recebe o pedido e, através das páginas PHP, processa-o e dá resposta. Os **clientes controladores** recebem a resposta e atuam as saídas em conformidade.

É possível controlar e monitorizar qualquer sistema através de uma interface *web* que está acessível a qualquer *cliente externo*. Este ao realizar o controlo de um periférico, por exemplo rodar o painel solar, envia um pedido ao servidor *web* para escrever na base de dados essa ordem, ordem essa que posteriormente será lida pelo **cliente controlador** e assim atuará as saídas de acordo com o pedido pelo **cliente externo** (utilizador).

A juntar a estas funcionalidades, foram desenvolvidos modos automáticos e manuais de controlo de todos os sistemas propostos, para que assim seja possível o sistema ser autónomo e adaptativo a cada tipo de utilizador, sendo que os parâmetros de funcionamento automático são inseridos pelo próprio.

No capítulo 4 irá ser abordado e apresentado todo o *Hardware* e *Software* utilizado e desenvolvido durante a realização da dissertação, assim como os constituintes de cada sistema.

Capítulo 4

Implementação

Neste capítulo será apresentada a implementação da solução proposta, isto é, irão ser mostrados os vários sistemas desenvolvidos e os seus propósitos. Primeiramente irá ser apresentado todo o *hardware* usado na implementação e depois o *software* utilizado e desenvolvido ao longo da realização da dissertação. Nesta última secção irá haver lugar à demonstração e explicação dos algoritmos desenvolvidos para os sistemas.

4.1 Sistema Proposto

Durante a realização da presente dissertação foram desenvolvidos vários sistemas que de seguida serão apresentados e devidamente enquadrados no âmbito da domótica.

Tal como referido no capítulo 1, domótica é automação no âmbito doméstico, o que leva o enquadramento da dissertação para a planta de uma habitação genérica. Na Figura 4.1 é possível observar uma habitação genérica, com quartos, casas de banho, cozinha, sala, *hall* e garagem.

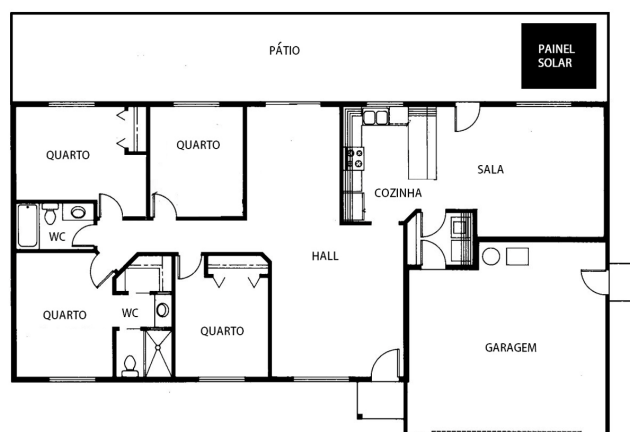


Figura 4.1: Planta genérica de uma habitação.

Considerando que a habitação genérica contém um sistema de AQS idêntico ao apresentado na Figura 2.11, é possível então controlar as recirculações de água nos circuitos solar térmico e outros, e monitorizar as temperaturas nos mesmos e no termoacumulador.

Tendo isso em mente, foi desenvolvido o sistema (**sistema água**) que pode ser visto na Figura 4.2. Este sistema é um dos sistemas propostos, monitoriza a temperatura dos diversos circuitos independentes, caldeira e solar térmico e a do termo-acumulador, e só promove a circulação de água nos circuitos independentes conforme promover a condução de calor positiva para a água do termoacumulador.

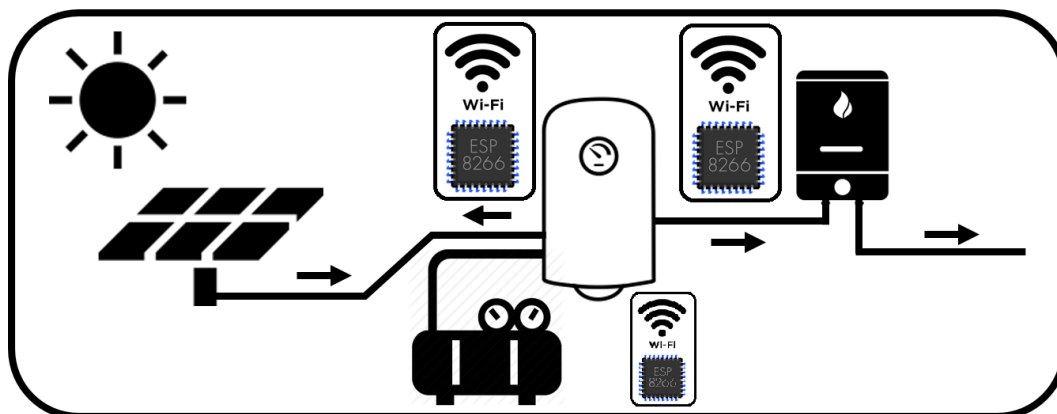


Figura 4.2: Sistema de monitorização e controlo de AQS acoplado com um **cliente controlador**.

O segundo sistema proposto (**sistema motor**) é baseado na rotação de um painel solar térmico para assim acompanhar o movimento do sol e maximizar o uso de energia solar para o aquecimento de águas. É possível monitorizar a posição atual do painel e controlar a sua posição através do modo manual ou automático. Na Figura 4.3 é possível, então, observar o esquema deste sistema.

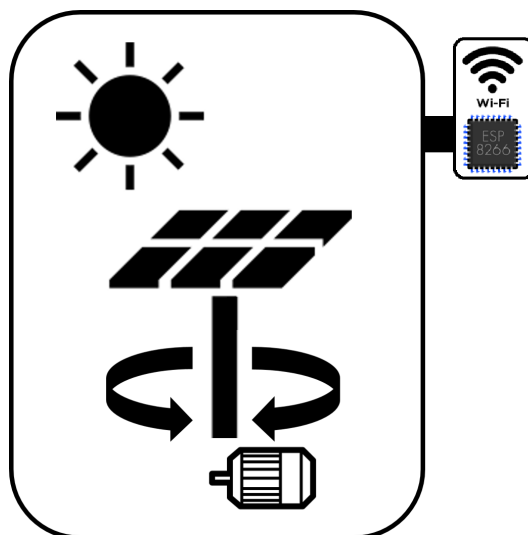


Figura 4.3: Sistema de monitorização e controlo da rotação de um painel solar acoplado com um **cliente controlador**.

O terceiro sistema proposto (**sistema ar**) é baseado no aquecimento do ar interior da habitação. Com este é possível monitorizar a temperatura no interior da habitação e

no exterior. É possível ativar a recirculação do ar interior, a introdução de ar novo na habitação, assim como realizar o aquecimento do ar. Na Figura 4.4 é possível observar um esquema ilustrativo.

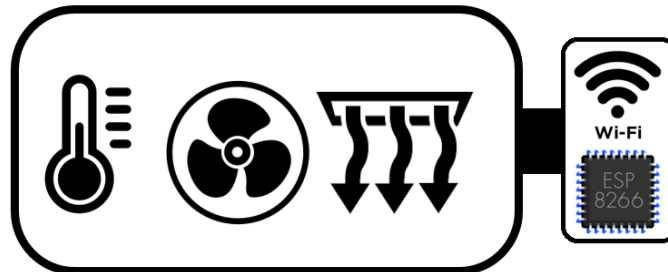


Figura 4.4: Sistema de monitorização e controlo do ar interior acoplado com um **cliente controlador**.

Tendo sido apresentados todos os esquemas exemplo dos sistemas desenvolvidos é pertinente fazer o enquadramento desses mesmos sistemas na planta genérica de uma habitação. Na Figura 4.5 é possível ver os três sistemas apresentados já enquadrados acoplados com os respetivos **clientes controladores**. De destacar o contorno amarelo, feito a uma parte da habitação, que indica a zona útil, isto é, todos os compartimentos que devem ser sujeitos a climatização porque têm uso constante/continuado pelos utilizadores da habitação.

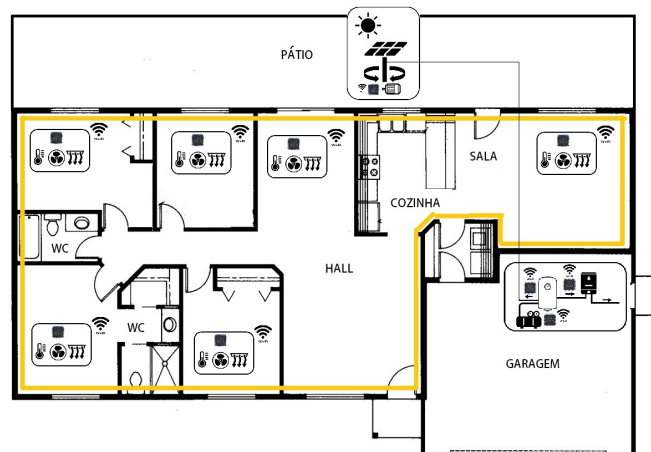


Figura 4.5: Planta genérica de uma habitação com os sistemas propostos instalados.

Para todo o sistema proposto estar completo, tem de ser acrescentado à Figura 4.5 o servidor, a rede local e o **cliente externo** (utilizador). Na Figura 4.6 foi feita essa adição. De destacar também a presença do meio de comunicação escolhido para realizar as comunicações, o *Wi-Fi* padrão, norma IEEE 802.11, isto é, uma versão sem fios do protocolo *Ethernet* [22].

Para controlar todo o sistema apresentado foi também desenvolvida uma interface que faz monitorização e controlo de todo o sistema, tal como foi referido no capítulo 3. Deste modo, através de um qualquer equipamento ligado à WLAN (*Wireless Local*

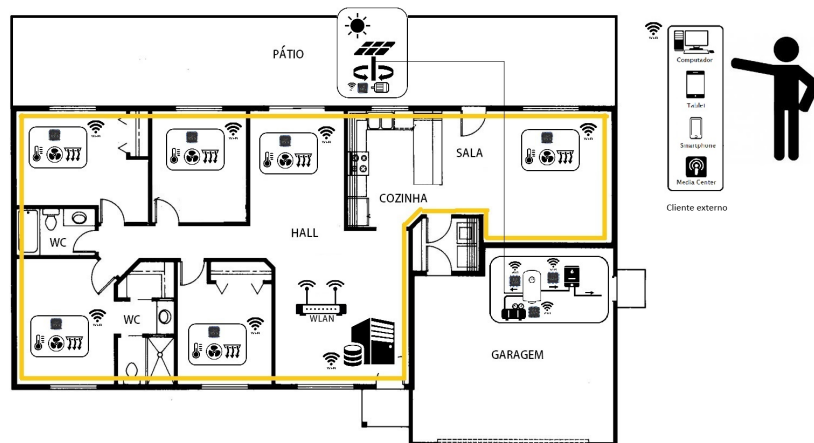


Figura 4.6: Planta genérica com os sistemas instalados, o servidor, o meio de comunicação e o **cliente externo**.

Area Network) que tenha um *browser* pode controlar e monitorizar os três sistemas apresentados. Na Figura 4.7 é possível observar possíveis ecrãs da interface quando visualizado através de um computador, é também possível aceder à interface através de um dispositivo que tenha um ecrã mais pequeno, um telemóvel por exemplo, estando a página preparada para tal. Mais informação sobre a interface desenvolvida presente no Apêndice B.

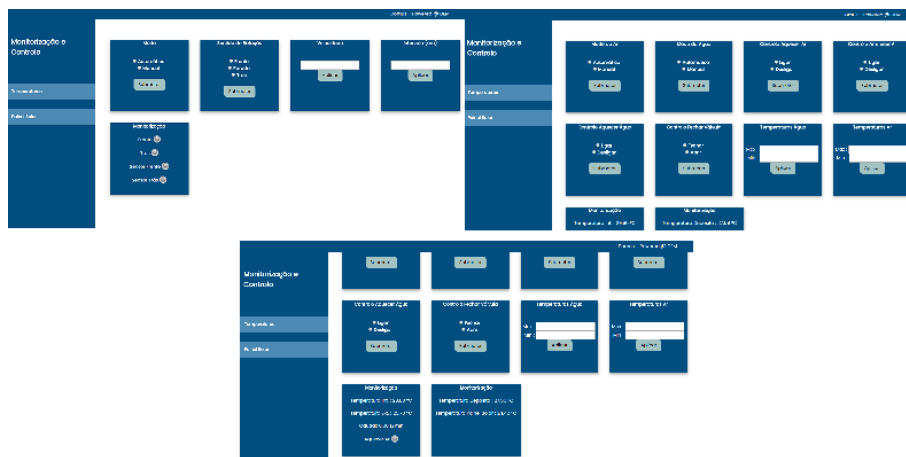


Figura 4.7: Várias janelas da interface *web*.

Os sistemas anteriormente apresentados poderiam ter sido idealizados sendo o **cliente controlador** um controlador que se conectava através de cabo *Ethernet* até ao servidor. No entanto devido a desenvolver um sistema mais versátil e de fácil acoplamento a todas as habitações, optou-se por uma solução *Wi-Fi*. Desta opção obtiveram-se ganhos em termos de conectividade e facilidade de integração nos sistemas já existentes. Na secção 4.2 irá ser abordado todo o *hardware* utilizado, sendo que o micro controlador *Esp8266* terá um destaque relevante.

4.2 Hardware

O *hardware* utilizado para desenvolver os **clientes controladores**, teve por base o micro controlador *Esp8266* que já integra um *shield wi-fi*. A juntar a este facto ainda pode ser programado em ambiente Arduino, o que ajuda à integração dos sistemas desenvolvidos. Além do *Esp8266* foi também usada em todos os sistemas desenvolvidos uma placa de relés para fazer a ativação de periféricos como resistências elétricas para o aquecimento de águas, ventoinhas para recirculação do ar, entre outros.

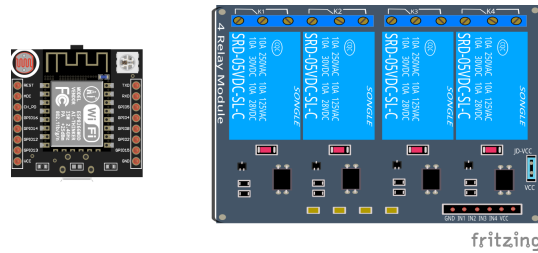


Figura 4.8: *Hardware* base do **cliente controlador**.

A estrutura da presente secção vai ser apresentar todos os equipamentos comuns no início e depois separar os componentes específicos de cada sistema por sub-secções.

4.2.1 *Esp8266*

Um elemento comum a todos os **clientes controladores** é o micro controlador *Esp8266*. Um *chip* criado pela Espressif e que tem como maior fator diferenciador o facto de ter um *shield wi-fi* incorporado.

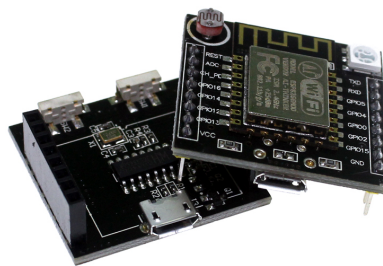


Figura 4.9: Placa de desenvolvimento que contém o *chip Esp8266 12F - Esp8266 MINI NODEMCU*.

Para a realização de todos os sistemas criados, foi usada uma placa de desenvolvimento que continha o *chip Esp8266 12F*. Na Figura 4.9 é possível ver a placa usada, de

referir que a placa usada é a que está por cima na imagem, a placa que está por baixo é o programador para transferir para o micro controlador o *software* desenvolvido.

Algumas características do *chip Esp8266 12F* podem ser enumeradas [23]:

- Suporta ligações através de *Wi-Fi* com a norma 802.11 b / g / n;
- Micro controlador com frequência de funcionamento de 80 MHz;
- Capacidade comunicar por protocolo TCP/IP, com bibliotecas dedicadas para o efeito;
- Possui porta de comunicação série (EIA 232), *I²C* (*Inter-Integrated Circuit*) entre outras;
- 16 pinos e um pino ADC (*Analog to Digital Converter*).

4.2.2 Placa de relés

Outro elemento comum a todos os **clientes controladores** foi uma placa de relés ativados a 5 *Volts*. Na Figura 4.10 é possível observar a placa em questão e o esquema do circuito eléctrico da mesma. De notar que a ativação dos relés é feita através de circuitos independentes que contêm um optoacoplador por relé, o que quer dizer que os circuitos são isolados uns dos outros e a parte de potência fica isolada do controlador que ativar o relé [12].

Em todos os sistemas foi usada uma placa destas, mas apenas utilizados, no máximo, três relés, o que indica que poderia ter sido usado placas de relés individuais ou de dois em certos casos.

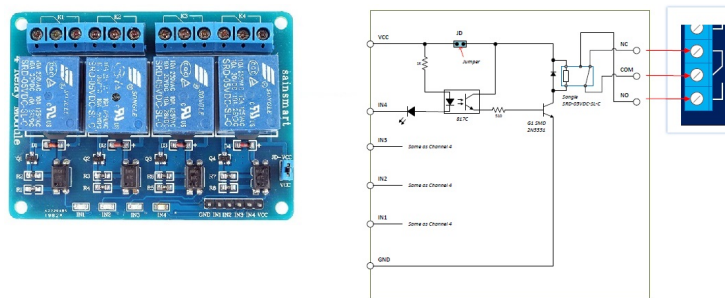


Figura 4.10: Placa de 4 relés [12].

4.2.3 Sistema Água - *Hardware*

Dando continuidade às descrições adotadas na secção 4.1 para a identificação dos vários sistemas tem-se que o **sistema água**, é um sistema que obtém dados de sensores de temperatura e de caudal e faz ativar eletroválvulas ou resistências eléctricas de modo a fazer recirculações de água nos circuitos independentes, solar térmico e caldeira, ou aquecer a água no termoacumulador com recurso a uma resistência.

Para realizar as tarefas enunciadas recorreu-se ao seguinte *hardware*:

- *Esp8266* MINI NODEMCU;

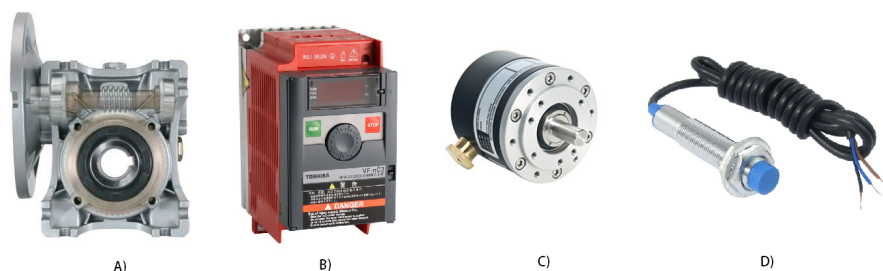


Figura 4.12: Alguns componentes utilizados no sistema motor. **A**-Redutor **B**-Variador **C**-Encoder **D**-Sensor Indutivo

Na Figura 4.12 pode ser visto os componentes mais caraterísticos deste sistema.

De notar que neste sistema foi utilizado mais que um *Esp8266* MINI NODEMCU devido ao facto de este ser uma placa de desenvolvimento e por isso já ter portas do chip destinadas a certas saídas e entradas digitais incorporadas na placa, o que faz com que tenham resistências já soldadas na placa para a ligação a LED's (*Light Emitting Diode*).

Neste sistema foram usados três *Esp8266* MINI NODEMCU, um que regista os impulsos vindos do *encoder*, outro que envia um sinal ao variador a indicar a velocidade de rotação do motor, regista um dos sensores de fim de curso e faz comutar os relés para o sinal do sentido de rotação, e o último tem o outro sensor de fim de curso.

4.2.5 Sistema Ar - *Hardware*

O **sistema ar**, o último, adquire dados de sensores de temperatura instalados no exterior e no interior de uma habitação, tal como mostrado em 4.1. Deste modo é possível fazer recircular o ar interior, introduzir ar novo no ambiente controlado e fazer o aquecimento do ar, tudo isto através de um modo automático ou de um modo manual. Para a realização do explicado anteriormente é necessário o seguinte *hardware*:

- *Esp8266* MINI NODEMCU;
- Placa de relés;
- ADS 1115, conversor analógico digital de 4 entradas que comunica por I^2C e tem uma resolução de 16 *Bits*;
- Sensor de temperatura LM35 CAZ;
- Termopar do tipo K.

A escolha do termopar recaiu sobre o termopar tipo K, pois este é o tipo mais indicado para aplicações genéricas. A gama de temperaturas permitida pelo mesmo é bastante ampla, no entanto as temperaturas a medir serão de uma gama muito mais baixa. O conjunto de metais usado neste termopar é o Cromel (90% Ni e 10% Cr) e o Alumel (95% Ni, 2% Mn, 1% Si e 2% Al). De referir ainda que este é o tipo de termopar mais usual.

4.2.6 Considerações gerais sobre o sistema proposto

Para o desenvolvimento de todos os sistema foi tido em consideração o isolamento de todos os circuitos, isto é, usando circuitos optoacoplados em que se separa, sempre, a parte de potência da parte do processamento de sinal e tratamento de dados, o *Esp8266*.

Os optoacopladores estão disponíveis com diversas configurações. Neste caso para o desenvolvimento do sistema havia à disposição dois tipos, o IL300 e o 4N25. Na Figura 4.13 podem ser vistos os circuitos dos dois.

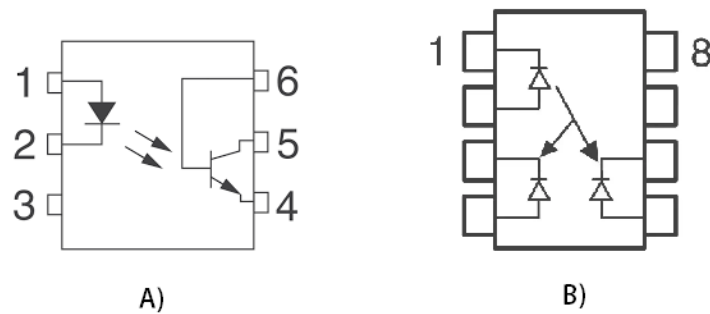


Figura 4.13: Optoacopladores. **A**-4N25 **B**-IL300

Neste caso, para o desenvolvimento dos sistemas, foram utilizados os optoacopladores 4N25. A escolha recaiu sobre estes dado que a utilização era para sinais digitais, logo a tipologia era ou 0 (zero) ou 1 (um) não sendo necessário ter uma variação linear ao longo de várias tensões de entrada. O optoacoplador IL300 possibilita que a tensão de saída seja proporcional à de entrada, como para a aplicação não era necessário, a escolha recaiu sobre o optoacoplador 4N25. Outra característica que pode influenciar a decisão é a questão monetária, sendo que o optoacoplador 4N25 é mais barato que o IL300.

4.3 Software

O *software* desenvolvido para os *Esp8266* dos sistemas apresentados na secção 4.1 foi desenvolvido em *Visual Studio*. Para tal ser possível, foi instalado um *plug-in*, o *Visual Micro*, para assim ser possível desenvolver *software* compatível com placas *Arduino* e semelhantes que suportem a linguagem. Na Figura 4.14 é possível ver o ambiente do *Visual Studio*.

A escolha recaiu sobre o *Visual Studio* para desenvolver o *software*, isto é devido a ser um programa já usado anteriormente numa unidade curricular e devido à facilidade de entender a interface e de escolher qual a placa a programar. Também todas as bibliotecas do *Arduino* são compatíveis com o *Visual Studio*.

A presente secção começará por apresentar e explicar o *software* comum aos três sistemas desenvolvidos e posteriormente apresentar o mais específico de cada um dos sistemas. No final, subsecção 4.3.5, será apresentado e explicado todo o desenvolvimento realizado no lado do servidor, como a base de dados e os ficheiros PHP. O que diz respeito à interface de monitorização e controlo pode ser encontrado no Apêndice B.

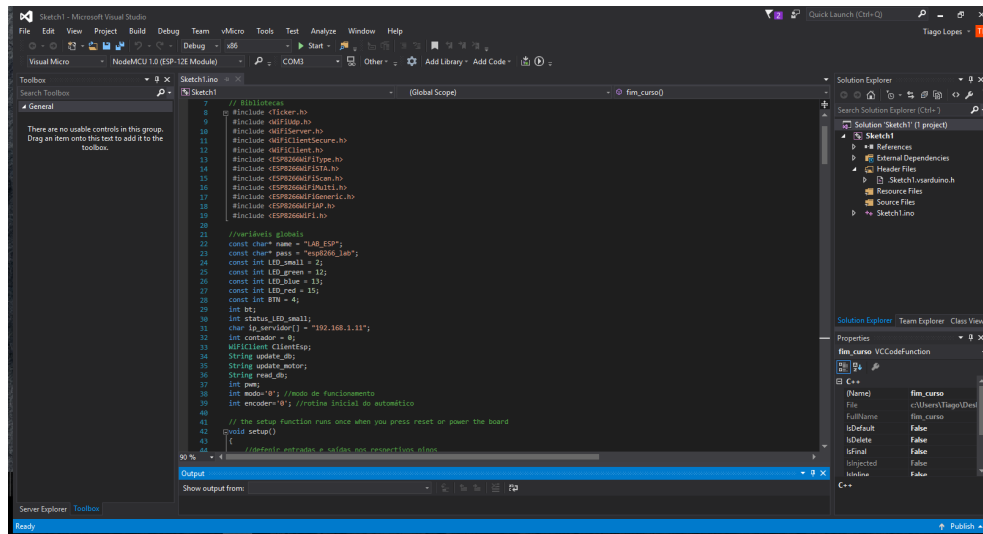


Figura 4.14: Janela do Visual Studio.

4.3.1 Desenvolvimento Comum

Para os *Esp8266* realizarem comunicações com o servidor têm de estar capacitados de fazer pedidos a este via *wi-fi*. Para tal poder acontecer foi utilizada uma biblioteca, *ESP8266WiFi.h*, que possibilita várias funcionalidades, entre elas aceder a uma rede *wi-fi* e enviar mensagens por TCP/IP a outros equipamentos conectados a essa mesma rede. Para isto ser possível é preciso saber o IP do equipamento de destino.

TCP/IP é um conjunto de protocolos que permite que vários dispositivos comuniquem entre si, estando todos ligados à mesma rede local. Este conjunto de protocolos tem várias camadas bem definidas, sendo que cada uma desempenha uma função no envio da mensagem. As camadas são: *Application Layer*, *Transport Layer*, *Internet Layer* e *Interface Layer*. Na Figura 4.15 é possível observar a organização das camadas com mais detalhe [13].

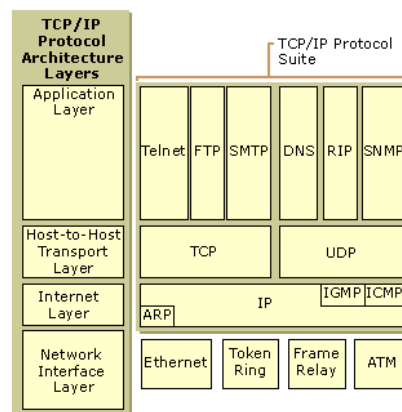


Figura 4.15: Organização das camadas do protocolo TCP/IP [13].

Na Figura 4.16 é possível ver o excerto de código que realiza a conexão do *Esp8266*

ao *router*. De notar que o nome do *router* ao qual se quer aceder é “LAB_ESP” e a palavra passe é “esp8266_lab”.

```
const char* name = "LAB_ESP";
const char* pass = "esp8266_lab";
WiFiClient ClientEsp;

WiFi.begin(name, pass);

while (WiFi.status() != WL_CONNECTED)
{
    status_LED_small = !status_LED_small;
    digitalWrite(LED_small, status_LED_small);
}
```

Figura 4.16: Código que estabelece a ligação do *Esp8266* ao *router*.

Terminada a ligação do *Esp8266* ao *router* é necessário que este se conecte ao servidor para realizar pedidos e receber respostas aos mesmos. Para tal, foi criada uma função que faz o envio de uma mensagem para o servidor pela porta 80 com um pedido de uma página PHP. Este pedido tem como parâmetros de entrada os valores instantâneos das variáveis que se pretendem armazenar na base de dados. O método utilizado para realizar o pedido é o método *GET*.

O método *GET* é um de dois métodos possíveis de serem utilizados para realizar pedidos de ficheiros PHP. Os dois métodos são o método *GET* e o método *POST*, neste caso a escolha recaiu sobre o *GET* devido aos dois intervenientes do pedido, o *ESP8266* e o servidor, não necessitarem de proteger as mensagens trocadas entre si, dado não haver interação humana nas mesmas. Se houvesse necessidade de proteção de dados seria utilizado o método *POST*, sendo este o mais indicado quando se trata de dados confidenciais.

Na Figura 4.17 é possível ver a função criada para a realização do pedido de páginas PHP ao servidor.

```
void update_database()
{
    //Ligar ao Servidor MySQL para poder escrever na base de dados
    ClientEsp.connect(ip_servidor, 80);

    if (ClientEsp.connected())
    {
        update_db = String("GET /tiago_tese/update_db.php?");
        update_db = update_db + "acumulador=" + termo;
        update_db = update_db + "&sensor=" + q;
        update_db = update_db + "&solar=" + analog;

        ClientEsp.println(update_db);
        //Serial.println(update_db); //Imprimir no terminal a frase que envia ao php

        read_db = ClientEsp.readString();
        //Serial.println(read_db); //Imprimir no terminal o que vem do php com o controlo
        controle();

        ClientEsp.stop();
        ClientEsp.flush();
    }
}
```

Figura 4.17: Função que envia um pedido de página PHP ao servidor, com o valor atualizado das variáveis a monitorizar.

Além de ser capaz de enviar para o servidor pedidos para escrever na base de dados, o *Esp8266* também tem de ser capaz de pedir informações à base de dados, isto é, ler os

valores de controlo atualizados por parte do utilizador (**cliente externo**).

Os ficheiros PHP podem ser programados para darem respostas com algumas variáveis presentes na base de dados. Deste modo, foi criada uma função que faz a leitura e interpretação das respostas recebidas. Na Figura 4.17 é possível observar que depois de ser realizado o envio do pedido para o servidor, com o comando

`ClientEsp.println(update_db)`, se faz a leitura da resposta que é retribuída, `read_db = ClientEsp.readString()`. Deste modo, a resposta recebida é armazenada na variável `read_db`, depois é chamada a função `controlo()` que realiza a interpretação e o tratamento da resposta. Tal pode ser observado na Figura 4.18.

```
void controlo()
{
    if (read_db.length() == 14)
    {
        //Modo
        if (read_db[11] == '1')
        {
            modo = 1;
        }
        else
        {
            modo = 0;
        }

        //Max ar
        max_agua = (((int)read_db[6] - 48) * 10 + ((int)read_db[7] - 48));

        //Min ar
        min_agua = (((int)read_db[8] - 48) * 10 + ((int)read_db[9] - 48));

        if (modo == 0)
        {
            //Fechar a valvula
            if (read_db[12] == '1')
            {
                digitalWrite(16, LOW);
            }
            else
            {
                digitalWrite(16, HIGH);
            }

            //Ligar a chaleira
            if (read_db[13] == '1')
            {
                digitalWrite(8, LOW);
            }
            else
            {
                digitalWrite(8, HIGH);
            }
        }
    }
}
```

Figura 4.18: Função que realiza a leitura e interpreta a resposta dada pela base de dados, através do ficheiro PHP.

4.3.2 Sistema Água - *Software*

Como visto anteriormente na secção 4.2.3 e na 4.1, o propósito deste sistema é aquecer a água num termoacumulador com as fontes de energia disponíveis, da forma mais eficiente possível; para tal foi desenvolvido um algoritmo de tomada de decisão baseado nos valores instantâneos medidos pelos sensores. Na Figura 4.19 é possível observar a função desenvolvida para o efeito, a seta a vermelho indica a continuidade do código.

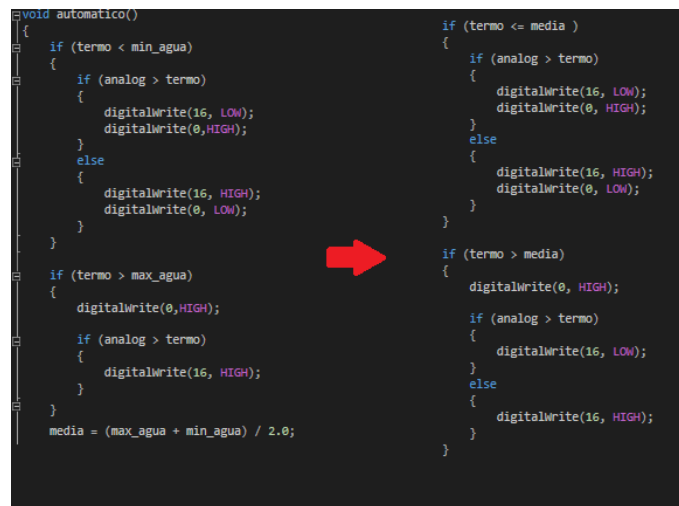
A função desenvolvida apenas é chamada quando o utilizador, **cliente externo**, opta pelo funcionamento em modo automático definindo um valor máximo e um valor mínimo para a temperatura da água.

Na Figura 4.19 a variável `min_agua` é a temperatura mínima definida pelo utilizador, `max_agua` é a temperatura máxima, `termo` é a temperatura dentro do termo-

acumulador e `analog` é a temperatura medida no circuito do sistema solar térmico. Com estes valores foi possível criar um algoritmo, de lógica booleana, de tomada de decisão.

A primeira condição `if` compara a temperatura do termoacumulador com a mínima definida pelo utilizador, se for menor dentro do termoacumulador não está de acordo com o especificado pelo utilizador, então entra dentro da condição, que por sua vez analisa a temperatura dentro do circuito solar térmico, se for maior dentro do circuito solar térmico que dentro do termoacumulador então é promovida a transferência de calor entre eles abrindo a eletroválvula (`digitalWrite(16, LOW)`) e desligando a resistência elétrica que aquece a água (`digitalWrite(0,HIGH)`). Se a temperatura for menor no circuito solar térmico que no termoacumulador então vai acontecer o contrário ao anterior, eletroválvula fechada para não promover trocas de calor com o solar térmico e resistência ligada promovendo o aquecimento da água. Se houvessem mais circuitos independentes, como caldeiras, bombas de calor, etc seria necessário replicar as condições `if` para esses casos.

A segunda condição `if` compara a temperatura do termoacumulador com a máxima definida pelo utilizador. Se a temperatura dentro do termoacumulador for maior do que a definida pelo utilizador, a resistência elétrica é desligada (`digitalWrite(0,HIGH)`). Posteriormente, já no interior da condição `if`, é realizada a comparação para averiguar se a temperatura do circuito solar térmico é maior que a do termoacumulador. Se for verificada que a temperatura do sistema solar térmico é maior que a do termoacumulador é fechada a eletroválvula e não se realiza transferência de calor, assim a água é mantida dentro dos limites pedidos pelo utilizador. Mais uma vez se houvessem mais circuitos independentes seria necessário replicar as condições `if` para esses casos.



```

void automatico()
{
    if (termo < min_agua)
    {
        if (analog > termo)
        {
            digitalWrite(16, LOW);
            digitalWrite(0,HIGH);
        }
        else
        {
            digitalWrite(16, HIGH);
            digitalWrite(0, LOW);
        }
    }

    if (termo > max_agua)
    {
        digitalWrite(0,HIGH);

        if (analog > termo)
        {
            digitalWrite(16, HIGH);
        }
    }

    media = (max_agua + min_agua) / 2.0;

    if (termo <= media)
    {
        if (analog > termo)
        {
            digitalWrite(16, LOW);
            digitalWrite(0, HIGH);
        }
        else
        {
            digitalWrite(16, HIGH);
            digitalWrite(0, LOW);
        }
    }

    if (termo > media)
    {
        digitalWrite(0, HIGH);

        if (analog > termo)
        {
            digitalWrite(16, LOW);
        }
        else
        {
            digitalWrite(16, HIGH);
        }
    }
}

```

Figura 4.19: Algoritmo desenvolvido para o modo automático do sistema água.

De seguida é calculada a média entre a temperatura máxima e a mínima para descobrir qual a temperatura média a que água deve estar. Partindo desse valor criam-se mais duas condições `if`, a primeira verifica se a temperatura dentro do termoacumulador é inferior ou igual à média, se for verificada entra dentro da condição onde existe outra condição que verifica se a temperatura do sistema solar térmico é maior que a do termoacumulador, se for verificada, abre a eletroválvula e desliga a resistência que

faz o aquecimento da água dentro do termoacumulador. A segunda condição verifica se a temperatura dentro do termoacumulador é maior que a média, se for entra dentro da condição, onde é desligada a resistência elétrica sendo depois testada outra condição, se a temperatura do sistema solar térmico é maior que do termoacumulador, se for verificada a eletroválvula fica aberta, se não for, permanece fechada.

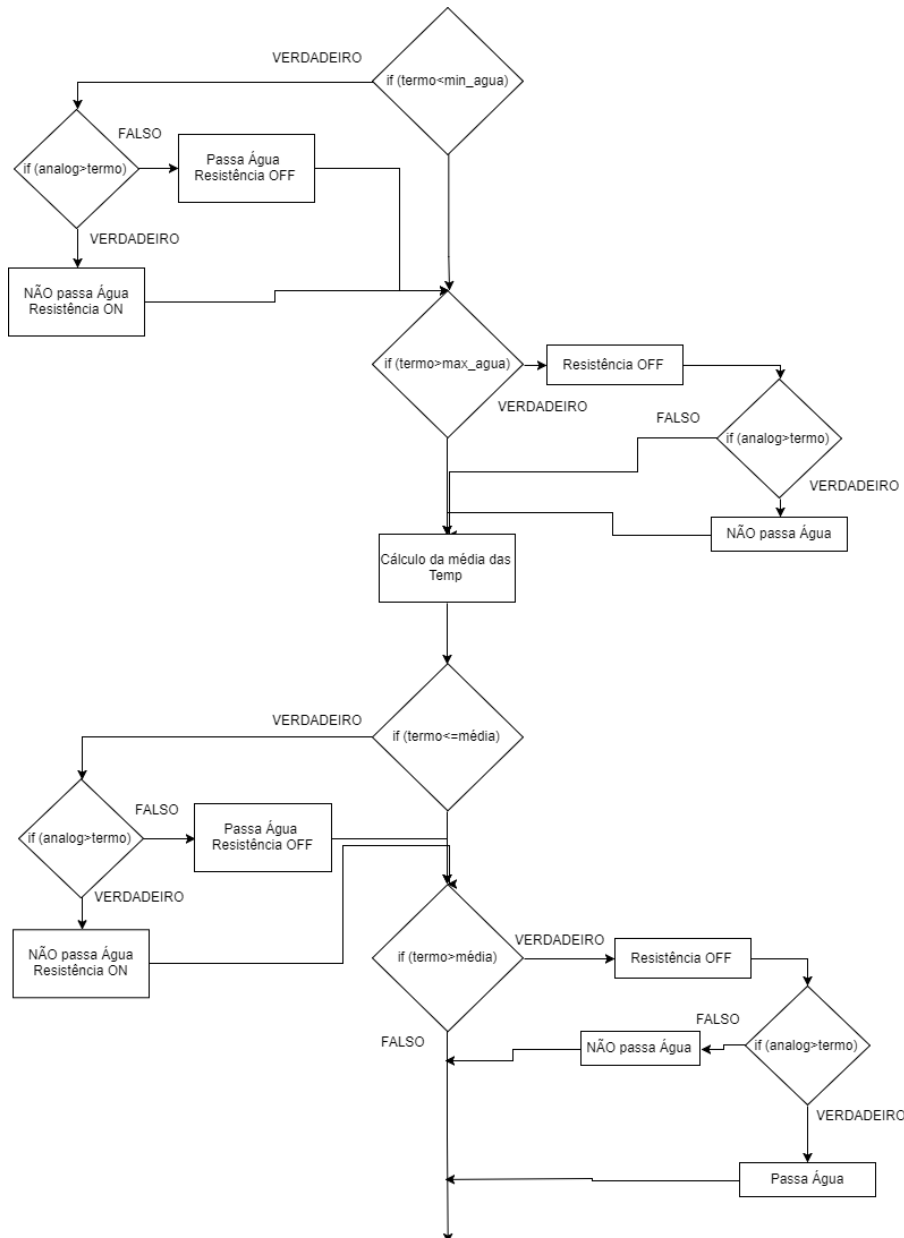


Figura 4.20: Diagrama do algoritmo desenvolvido para o sistema água.

4.3.3 Sistema Motor - *Software*

O sistema motor, tal como referido em 4.1 e em 4.2.4 tem como objetivo fazer rodar o painel solar de forma automática ou manual, isto é, definir um certo número de graus para rodar de um certo tempo em um certo tempo, no modo automático. Para realizar esta operação foi desenvolvido um algoritmo que faz sozinho esta operação. Na Figura 4.21 está o código desenvolvido que faz executar o modo automático, de notar que o *Esp8266* que contém esta função é o que está conectado ao *encoder*.

```

void automatico()
{
  if ((encoder == '1') && (tempo_parado==0))
  {
    graus = (10 * 23.5979); //10cm
    ClientEsp.connect(ip_servidor, 80);

    if (ClientEsp.connected())
    {
      ClientEsp.println(String("GET /tiago_tese/update_motor.php?stf=1&str=0"));
      //Serial.println(update_motor); //Imprimir no terminal a frase que envia ao php

      ClientEsp.stop();
      ClientEsp.flush();
    }
  }

  if ((encoder == '1') && (tempo_parado >= 10))
  {
    tempo_parado = 0;
    parado = 0;

    ClientEsp.connect(ip_servidor, 80);

    if (ClientEsp.connected())
    {
      ClientEsp.println(String("GET /tiago_tese/update_motor.php?stf=1&str=0"));
      //Serial.println(update_motor); //Imprimir no terminal a frase que envia ao php

      ClientEsp.stop();
      ClientEsp.flush();
    }
  }

  if ((encoder == '1') && (tops_encoder >= graus) && (graus != 0))
  {
    ClientEsp.connect(ip_servidor, 80);

    if (ClientEsp.connected())
    {
      ClientEsp.println(String("GET /tiago_tese/update_motor.php?stf=0&str=0"));
      //Serial.println(update_motor); //Imprimir no terminal a frase que envia ao php

      ClientEsp.stop();
      ClientEsp.flush();
    }

    tops_encoder = 0;
    parado = 1;
  }
}

```

Figura 4.21: Algoritmo desenvolvido para o modo automático do sistema motor.

A função desenvolvida apenas é chamada quando o utilizador utiliza o modo automático. A primeira condição `if` faz o cálculo de quanto é suposto rodar o veio do motor e envia um pedido ao servidor para atualizar na base de dados o estado do motor, isto é, rodar no sentido dos ponteiros do relógio. Para tal acontecer é enviada uma mensagem ao servidor, como visto em 4.3.1, mas tendo uma estrutura fixa, “GET /tiago_tese/update_motor.php?stf=1&str=0”. Deste modo é atualizado na base de dados este valor e o *Esp8266* que está preparado para interpretar esta resposta irá ler o campo e accionar o relé correspondente ao movimento de rodar para a direita. Irá ser explicado primeiro a terceira condição devido à ordem lógica do processo. Enquanto o número de impulsos (`tops_encoder`) não for igual ou maior ao número de graus definido para a rotação do eixo este não entra nesta condição, quando o for entra. Deste modo, ao entrar na condição é enviada um pedido ao servidor a cancelar o movimento de rotação para os dois lados, e dá-se o valor “1” à variável “parado”, significando que o sistema se encontrada parado e reinicia-se o contador do número de impulsos do *encoder*. Estando

o valor de “parado” igual a 1, existe uma rotina de acesso à interrupção que corre a cada segundo e que vai incrementando uma variável que tem como utilidade contar o tempo que o sistema está parado para efetuar outro movimento igual ao primeiro. A segunda condição do algoritmo é satisfeita quando um tempo definido é alcançado, neste caso quando o contador chega ao valor de 10. Depois é reiniciado o contador e a variável “parado” é posta a “0”. Por consequência o veio volta a rodar para o mesmo sentido que anteriormente, deste modo garante-se que o veio gira um certo número de graus de tempo a tempo.

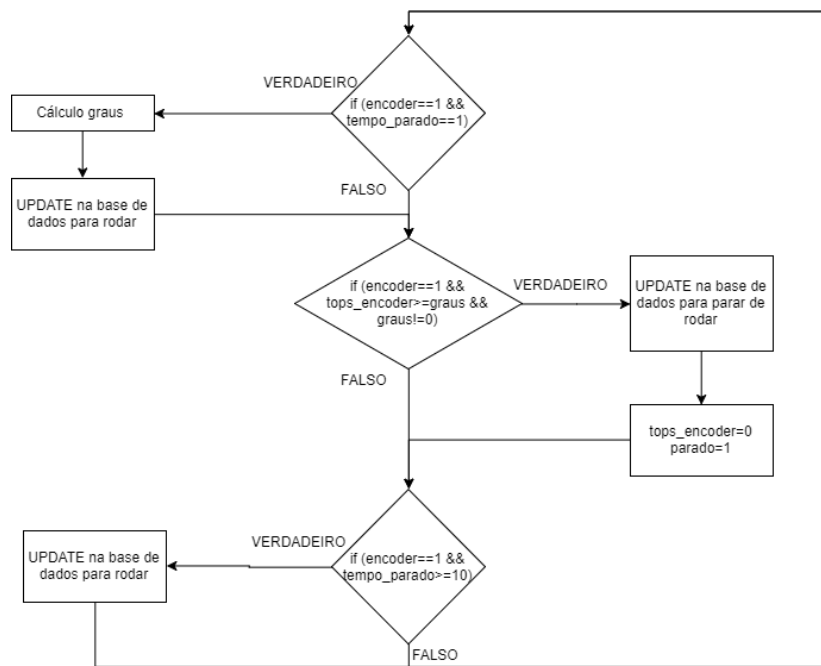


Figura 4.22: Diagrama do algoritmo desenvolvido para o sistema motor.

Ao ser detetado o fim de curso do lado para onde roda o veio, este volta ao fim de curso da ponta oposta sem interrupções. Este trecho de código está presente noutra *Esp8266*, no que contém a ativação dos relés para os dois sentidos. Na Figura 4.23 é possível observar, no trecho de código superior, que quando a entrada 5 do *Esp8266* estiver a “LOW” é porque o sensor de fim de curso detetou e então vai fazer parar o motor e vai atribuir o valor “0” à variável “encoder”, que só tem o valor de zero quando o eixo está a rodar para a posição de partida. Posteriormente a função que tem o nome “automático”, que também é visível na Figura 4.23, vai atualizar na base de dados que o motor vai rodar em sentido contrário e vai ativar o relé para que este rode no sentido contrário. Quando o sensor de fim de curso da posição inicial detetar presença vai alterar o valor da variável “encoder” para “1” e assim dar início ao ciclo de novo.

De salientar que neste sistema o **cliente controlador** é constituído por mais que um *Esp8266*, tal como referido em 4.2.4.

```

if (modo == '1')
{
    if (encoder == '1')
    {
        if (digitalRead(S) == LOW) //detecção sensor da frente
        {
            if (digitalRead(14) == LOW)
            {
                digitalWrite(14, HIGH);
                digitalWrite(0, HIGH);

                ClientEsp.connect(ip_servidor, 80);
                if (ClientEsp.connected())
                {
                    ClientEsp.println(String("GET /tiago_tese/update_motor.php?stf=0&str=0&encoder=0"));
                    //Serial.println(update_motor); //Imprimir no terminal a frase que envia ao php
                    ClientEsp.stop();
                    ClientEsp.flush();
                }
            }
            encoder = '0';
        }
    }
}

void automatico()
{
    if ((modo == '1') && encoder == '0')
    {
        ClientEsp.connect(ip_servidor, 80);
        if (ClientEsp.connected())
        {
            ClientEsp.println(String("GET /tiago_tese/update_motor.php?str=1&stf=0&pwm=100"));
            //Serial.println(update_motor); //Imprimir no terminal a frase que envia ao php
            ClientEsp.stop();
            ClientEsp.flush();
        }
        digitalWrite(0, LOW); //para trás
        digitalWrite(14, HIGH); //não para a frente
    }
}

```

Figura 4.23: Detecção de um sensor de fim de curso em modo automático

4.3.4 Sistema Ar - *Software*

O sistema ar é muito semelhante ao sistema água, mas aplicado ao aquecimento do ar no interior de uma habitação. Com este sistema é possível aquecer o ar, fazer recircular o ar, e também introduzir ar novo na habitação, tendo por base uma temperatura máxima e uma mínima que o utilizador pode escolher através da interface. Na Figura 4.23 é possível observar a função desenvolvida que atua no modo automático.

A primeira condição da função diz respeito ao valor mínimo dentro do espaço a controlar. Se a temperatura medida for menor que a definida pelo o utilizador, então irá ser ativada a saída número zero, que irá ligar um aquecedor elétrico que vai aquecer o ar. A segunda condição é para a temperatura máxima definida pelo utilizador, se a temperatura dentro do espaço a controlar for maior que a definida como máxima pelo utilizador, entra dentro da condição e desliga a saída zero, o aquecedor, liga o sistema para fazer a recirculação do ar na sala e se a temperatura exterior for mais baixa que a definida como máxima pelo utilizador, então insufla ar exterior para o interior.

De seguida foi calculada a média entre os dois valores, máximo e mínimo, dados pelo utilizador e o intervalo entre eles. Posteriormente foi feita uma redução do intervalo dado de modo a manter a temperatura sempre dentro de um intervalo mais pequeno do que o que é dado pelo utilizador. Na Figura 4.25 é possível observar o dito. A primeira condição a seguir aos cálculos atua de modo a tirar a temperatura da zona a vermelho, para tal liga-se o aquecedor, desliga-se a recirculação e se a temperatura do exterior for menor que a interior não se liga a insuflação de ar novo, se for maior faz-se insuflação. A segunda condição diz respeito a sair da zona azul, para tal liga-se a recirculação do ar, desliga-se o aquecimento e se a temperatura exterior for menor que a interior faz-se insuflação, se for maior não se faz. A terceira condição é quando está na zona verde, tem o aquecimento, a insuflação e recirculação de ar desligada.

```

void automatico()
{
  if (analog < min_ar) //Liga o aquecedor
  {
    digitalWrite(0, HIGH);
    digitalWrite(16, HIGH);
    digitalWrite(LED_green, HIGH);
  }

  if (analog > max_ar) //Liga a ventoinha
  {
    digitalWrite(0, LOW);
    if (termo < max_ar)
    {
      digitalWrite(16, LOW);
    }
    else
    {
      digitalWrite(16, HIGH);
    }
    digitalWrite(LED_green, LOW); //sensação de arrefecimento devido ao circular do ar
  }

  intervalo = max_ar - min_ar;
  media = (max_ar + min_ar) / 2.0;

  if (analog <= (media - (0.2 * intervalo))) //LIGAR AQUECIMENTO
  {
    digitalWrite(0, HIGH);
    digitalWrite(16, HIGH);
    if (termo < analog)
    {
      digitalWrite(16, HIGH);
    }
    else
    {
      digitalWrite(16, LOW);
    }
  }

  if (analog >= (media + (0.2 * intervalo))) //LIGAR VENTONHA
  {
    digitalWrite(0, LOW);
    digitalWrite(16, LOW);
    if (termo < analog)
    {
      digitalWrite(16, LOW);
    }
    else
    {
      digitalWrite(16, HIGH);
    }
  }

  if ((analog < (media + (0.2 * intervalo))) && (analog > (media - (0.2 * intervalo))))
  {
    digitalWrite(0, LOW);
    digitalWrite(16, HIGH);
    digitalWrite(16, LOW);
  }
}

```

Figura 4.24: Algoritmo desenvolvido para o modo automático do sistema ar.

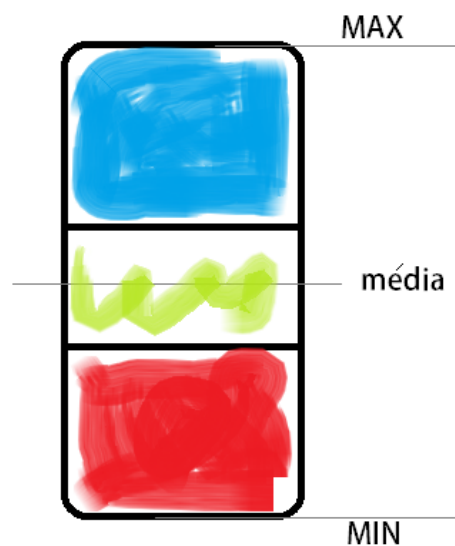


Figura 4.25: Esquema ilustrativo da lógica utilizada para desenvolver o algoritmo do sistema ar.

4.3.5 Servidor - *Software*

Para desenvolver o servidor foi usado o programa *XAMPP* que, entre outras coisas, possui a capacidade de ser servidor *web* e servidor base de dados. Isto é, possui a capacidade de fornecer páginas web a quem estiver ligado à mesma rede local que este. O servidor base de dados contém apenas a base de dados.

Na Figura 4.26 pode ser vista a interface do *XAMPP*. O primeiro módulo, denominado de *Apache*, é um servidor *web* e o segundo é um servidor base de dados, *MySQL*. *Apache* é o servidor *web* mais utilizado no mundo devido a ser de um *software* de livre acesso. O *MySQL* é um sistema que faz a gestão de dados e utiliza a linguagem SQL (*Structured Query Language*). Este sistema é, também, um dos mais usados do mundo para o armazenamento de dados.

No sistema proposto os clientes apenas comunicam com o servidor *web*, o servidor base de dados apenas recebe atualizações através do servidor *web*. Quando um **cliente controlador** envia um pedido ao servidor *web* para atualizar uma determinada grandeza na base de dados, este está a fazer um pedido de uma página PHP ao servidor *web* e a enviar certos campos já predefinidos que depois do lado do ficheiro PHP são interpretados e completados para serem enviados para o servidor base de dados já em linguagem SQL.

O **cliente externo**, o utilizador, apenas faz o pedido da página de controlo e monitorização, interface, ao servidor *web*, depois todo o controlo que este realizar é o envio de pedidos de páginas PHP que são depois transformados em *query's* SQL para alterar variáveis e grandezas na base de dados. A monitorização que existe na interface é com um princípio idêntico mas recorrendo também a ficheiros de *Java Script* além das páginas PHP.

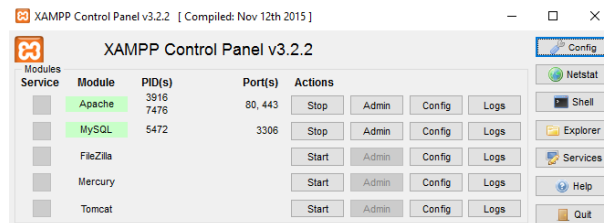


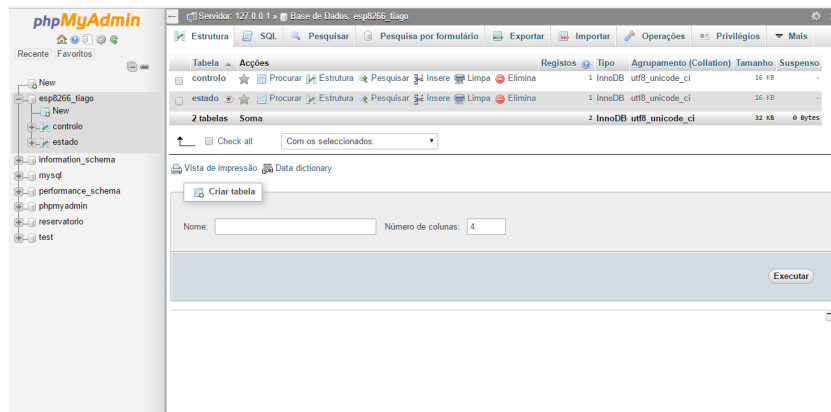
Figura 4.26: Interface do *software Xampp*.

Para a criação das bases de dados foi utilizada a interface *phpMyAdmin*, uma interface que está incluído na instalação do *XAMPP* e que é uma página PHP que se acede através de um *browser* e que serve de ferramenta para criação de bases de dados e gestão das mesmas. Na Figura 4.27 é possível observar que a interface é bastante intuitiva e agradável para aprender e trabalhar com a mesma.

Para todos os sistemas desenvolvidos deve ser criada uma base de dados para que seja possível armazenar informação relativa aos dados da habitação (temperatura do ar em cada divisão, temperatura da água dentro do termoacumulador e posição do painel solar) e também criar um histórico de valores para as temperaturas medidas.

Para satisfazer esta necessidade foi criada, então, uma única base de dados que foi depois normalizada, obtendo-se as seguintes tabelas:

- Solar térmico;
- Termoacumulador;

Figura 4.27: Interface *phpMyAdmin*.

- Movimento painel;
- Ar interior;
- Histórico termoacumulador;
- Histórico ar.

A tabela Solar térmico diz respeito a todos os dados de sensores de temperatura e de caudal que podem ser retirados do painel solar, também contém o estado de saídas como a recirculação no circuito. Pode ser observada a tabela criada na Tabela 4.1. De realçar a coluna existente com o nome *#Painel* para se houver mais que um sistema solar térmico ser possível ter todos os dados na mesma tabela, evitando repetições e tabelas duplicadas dentro da base de dados.

Tabela 4.1: Tabela Solar térmico

#Painel	T Out Painel	Caudal	Recirculação
1	60	6.3	1
...

A tabela Termoacumulador trata de todos os sensores de temperatura possíveis de utilizar no termoacumulador, contém também o estado de saídas como a ativação da resistência elétrica de aquecimento da água e tem, ainda, variáveis para auxiliar o bom funcionamento do sistema, isto é, o modo de funcionamento (coluna “Modo”) e os valores máximos e mínimos para o funcionamento automático do sistema que estão representados nas colunas “T Max” e “T Min”. É possível ver na Tabela 4.2 a tabela criada para o efeito.

Tabela 4.2: Tabela Termo acumulador

#Termo	T Out	T In	Resistência	T Max	T Min	Modo
1	60	20	0	65	50	1
...

A tabela Movimento painel armazena os valores de tudo o que seja relativo à rotação do painel solar, isto é, os sensores de fim de curso, a velocidade, o sentido de rotação,

o curso pedido, o modo de funcionamento e a posição atual do painel. Deste modo foi obtida a Tabela 4.3 . Nesta tabela a coluna *#Painel* diz respeito ao número do painel, podendo assim haver mais que um painel com este sistema evitando-se assim o repetir de tabelas e dados.

Tabela 4.3: Tabela Movimento painel

#Painel	STF	STR	Vel	Fim curso F	Fim curso R	Curso	Modo	Pos
1	1	0	100	0	0	19	0	16
...

A tabela Ar interior trata dos valores do sistema ar, temperatura interior da divisão a controlar, temperatura exterior, recirculação do ar interior, aquecimento, introdução de ar novo, modo de funcionamento e temperatura máxima e mínima. De salientar a coluna *#Div* que diz respeito à divisão que se está a monitorizar e controlar, assim é possível na mesma tabela escrever todos os dados relativos aos vários sistemas ar sem duplicar dados na base de dados. Isto é, supondo que a divisão número 1 é a sala e a divisão 2 é um quarto, é possível fazer o controlo e monitorização individual da temperatura das duas divisões a controlar. Havendo mais divisões para controlar e monitorizar será só inserir linhas com os devidos números correspondentes às divisões. A Tabela 4.4 exemplifica de forma clara o dito anteriormente.

Tabela 4.4: Tabela Ar interior

#Div	T In	T Out	Aquecer	Recirc	Ar novo	T Max	T Min	Modo
1	20	27	0	1	0	21	19	1
2	19	27	0	1	0	21	18	1
...

A tabela Histórico termoacumulador tem como objetivo registar os valores da temperatura dentro do termo acumulador ao longo tempo. Na Tabela 4.5 é possível observar a tabela criada, de salientar o facto de a coluna *Data* ser preenchida com a data e a hora do servidor, sendo assim definido esse atributo a quando da criação da tabela.

Tabela 4.5: Tabela Histórico termo acumulador

Data	Temperatura
2017-04-24 15:55:23	55
2017-04-24 15:56:23	56
...	...

A tabela Histórico ar é idêntica à tabela Histórico termoacumulador e tem o mesmo fim, registar o histórico da temperatura do ar no interior das divisões. Como pode haver mais que uma divisão esta tabela deve ser criada o número de vezes quantas divisões se estão a controlar e monitorizar, ou seja tem de haver tantas tabelas Histórico ar quantas linhas houver na tabela Ar interior. Na Tabela 4.6 é possível observar então uma tabela criada para uma das divisões.

Tabela 4.6: Tabela Histórico ar

Data	Temperatura
2017-04-08 10:35:53	19
2017-04-08 10:36:53	20
...	...

Capítulo 5

Análise de Desempenho

O presente capítulo tem como objetivo realizar a avaliação do trabalho desenvolvido, isto é, avaliar se a solução e o sistema proposto são viáveis.

Para realizar a análise de desempenho do sistema foram criados 3 casos de estudo, em que cada caso de estudo representa um sistema desenvolvido. Deste modo foi possível avaliar a solução proposta.

O presente capítulo será organizado em duas secções. A primeira secção abordará os casos de estudo criados e o porquê dos mesmos. A segunda secção será para realizar uma análise crítica dos casos de estudo.

5.1 Casos de Estudo / Protótipos Laboratoriais

Para ser possível testar os sistemas desenvolvidos foram criados 3 casos de estudo, um para cada sistema desenvolvido. Cada caso estudo foi pensado para aproximar o mais possível as condições reais das de teste, para deste modo ser possível realizar uma avaliação de funcionamento dos mesmos de maneira imparcial.

Para o **sistema água**, o de controlo e monitorização do aquecimento de águas sanitárias, foi montado o sistema visível na Figura 5.1. É possível observar nesta imagem que o sistema possível de ser usado em laboratório consiste em 2 chaleiras de água quente, uma eletroválvula e sensores de temperatura para cada uma das chaleiras.

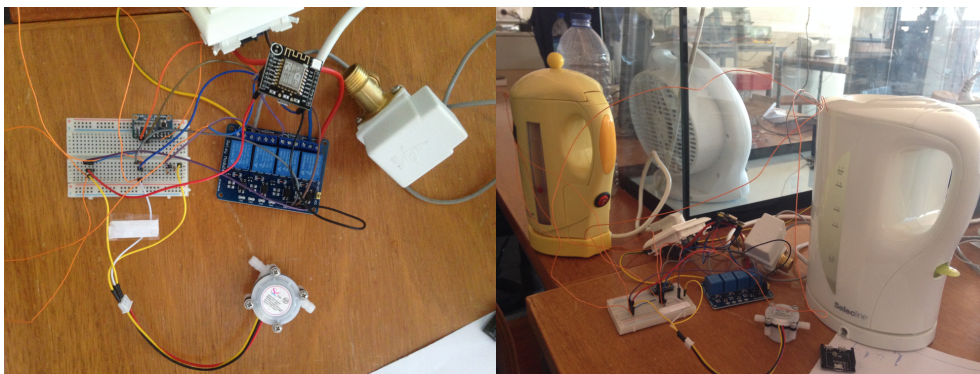


Figura 5.1: Caso de estudo para o sistema água.

Uma das chaleiras utilizada faz a simulação de um termoacumulador que tem uma

resistência elétrica para aquecer a água se necessário, a outra funciona para simular o circuito solar térmico. A que simula o termoacumulador é ativada através da programação do *Esp8266*, enquanto que a que simula o sistema solar térmico é usada em modo manual para realizar os testes de performance. A eletroválvula utilizada simula o corte de circulação de água no circuito solar térmico. Para saber se existe algum problema com o circuito de recirculação do sistema solar térmico foi também instalada um caudalímetro.

Para o **sistema motor**, o sistema que faz rodar o painel solar, foi utilizado um tapete com um sensor de fim de curso em cada extremidade. Deste modo simula-se a rotação do painel de uma forma realista e é também obtido o sinal de fim de curso de cada sentido de rotação. Na Figura 5.2 é possível observar o sistema montado para a realização dos testes.

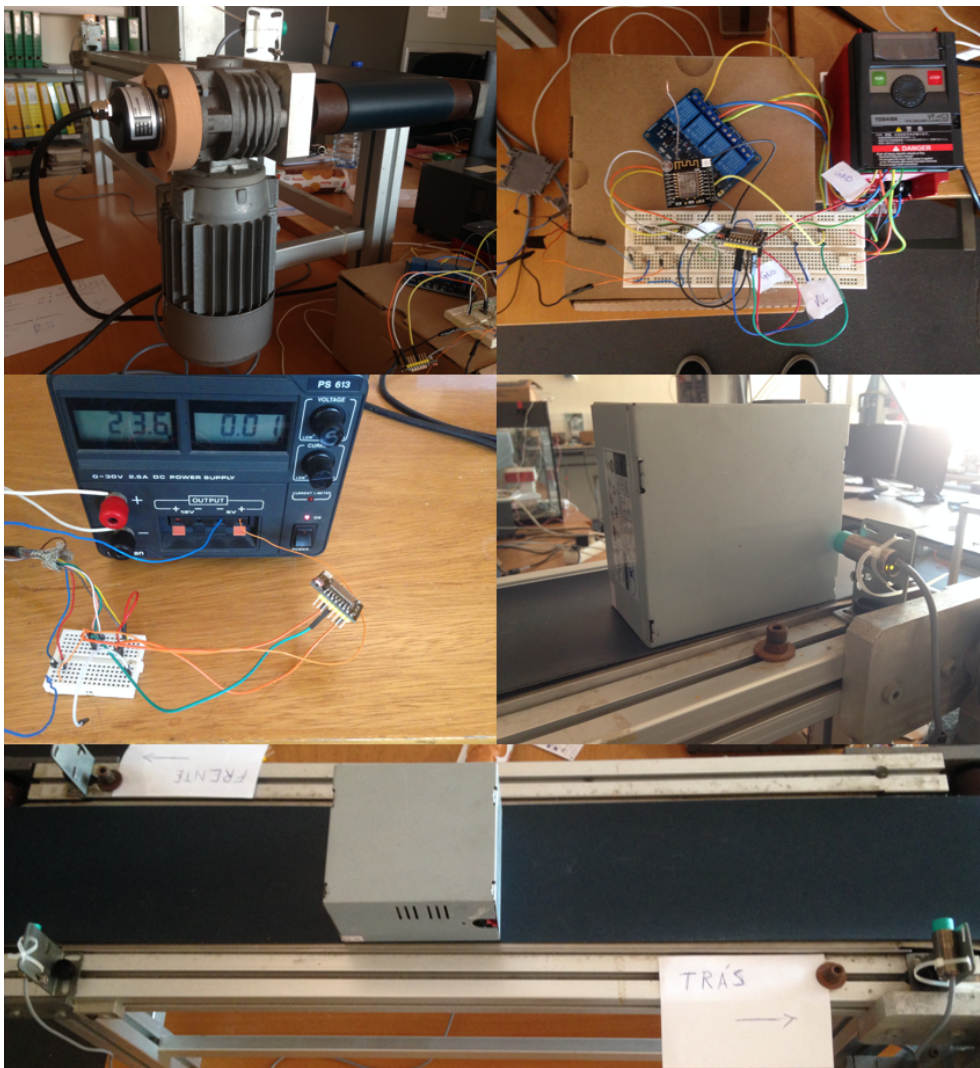


Figura 5.2: Caso de estudo para o sistema motor.

Para o último sistema, **sistema ar**, o controlo e monitorização da temperatura dentro da habitação, foi simulado, através de um ambiente controlado, o aquecimento, a recirculação de ar e a entrada de ar novo no espaço a controlar. Para tal, dentro do

ambiente controlado, foi colocado um sensor de temperatura, duas ventoinhas de 12 *volt* e um ventilador comum. Deste modo, o ventilador funciona de elemento que aquece o ambiente controlado, uma das ventoinhas é o elemento que faz recircular o ar interior e a outra ventoinha é o elemento que faz introdução de ar novo no ambiente controlado. Existe ainda um sensor de temperatura que mede a temperatura no interior do espaço a controlar e outro sensor que faz a medição no exterior.

A montagem realizada pode ser observada na Figura 5.3.

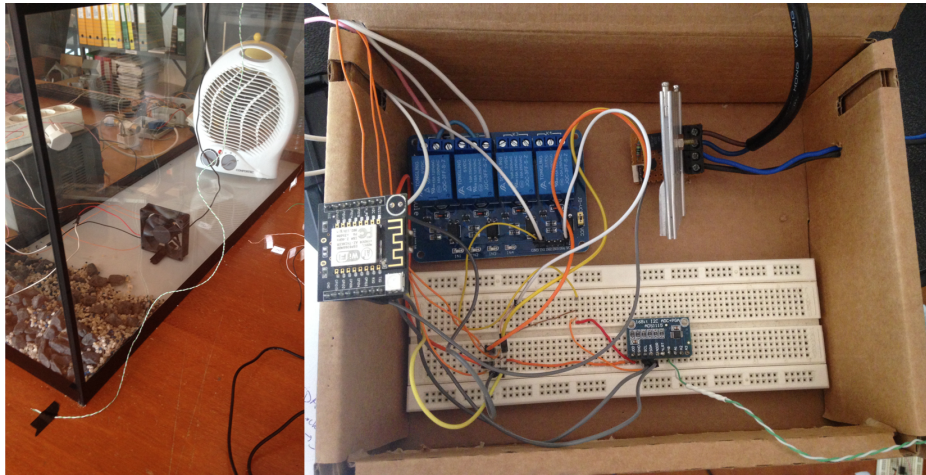


Figura 5.3: Caso de estudo para o sistema ar.

As três montagens apresentadas anteriormente são os casos de estudo possíveis de realizar em laboratório. Estes foram concebidos para simular, da melhor maneira, os casos reais onde seriam possíveis instalar os sistemas desenvolvidos. Deste modo, foi possível executar uma avaliação de desempenho do *software* e do *hardware* desenvolvido como se de um sistema real se tratasse. Aliado a estes sistemas foi também desenvolvida uma interface *web* para que fosse possível controlar e monitorizar o sistema de qualquer parte da habitação. Na Figura 5.4 é possível observar essa mesma interface. No Apêndice B é possível ver mais em detalhe todos os separadores e funcionalidades da mesma.

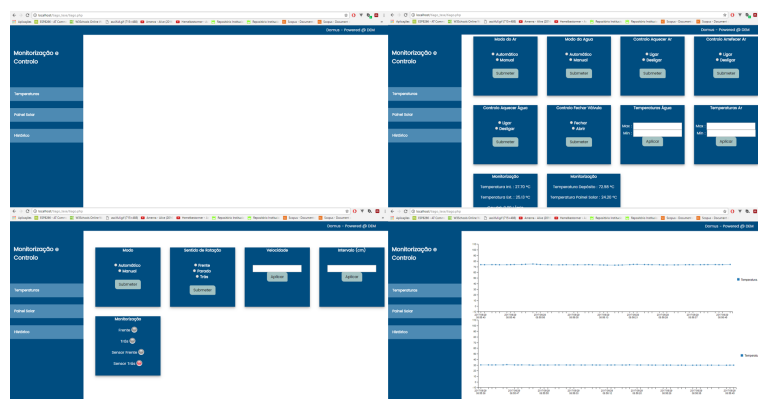


Figura 5.4: Várias janelas da interface desenvolvida para controlo e monitorização.

5.2 Análise de Desempenho dos casos de Estudo

Na presente secção irá ser abordado o desempenho dos casos de estudo apresentados anteriormente, destacando a precisão de medição dos sensores e o correto funcionamento dos modos de funcionamento, manual e automático.

O caso de estudo criado para o **sistema água** foi o apresentado na secção anterior. Da utilização deste pode-se observar que os sensores de temperatura utilizados nas duas chaleiras mediam um valor exato, isto é, a temperatura lida pelo sensor era idêntica à real. A temperatura real foi obtida por um termopar acessório de um multímetro, este já está calibrado para ser usado, assim irá medir uma temperatura exata. O sensor que está ligado à entrada analógica da placa de desenvolvimento media o valor idêntico ao do multímetro, variando uma décima de certo em certo tempo. O medido pode ser observado na Figura 5.5. A variação registada pode ser também devido ao erro do sensor (LM35), cerca de meio grau, segundo o fabricante do mesmo.

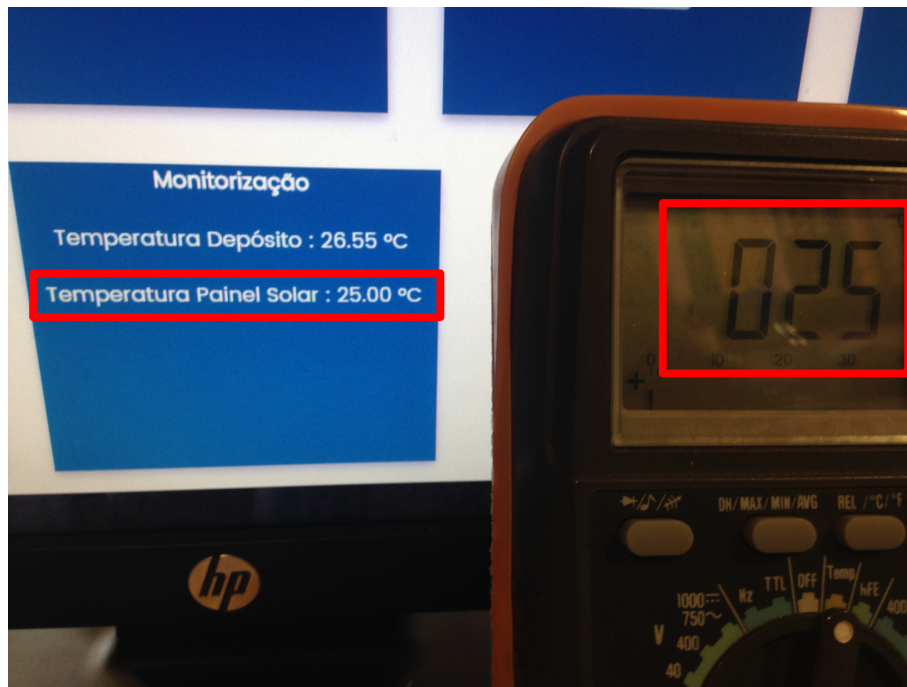


Figura 5.5: Comparação do valor medido com o LM35 conectado à ADC do *Esp8266* e com o sensor do multímetro.

O sensor de temperatura que estava conectado através da placa ADS 1115 fazia a leitura do valor com algum erro, cerca de meio grau. Na Figura 5.6 é possível observar esse mesmo erro. A fonte do mesmo pode ser devido à placa de conversão analógico digital ter uma resolução muito elevada (16 *bits*) e assim o valor medido poder conter algum ruído, também se pode dever ao erro do próprio sensor, tal como referido anteriormente. No entanto para a precisão desejada este erro é aceitável.

No modo automático observa-se que o sistema comporta-se de acordo com o previsto, sendo que a ativação dos relés ocorre nos limites previamente definidos e explicados em 4.3.2.

O caso de estudo do **sistema motor** foi o mais complexo de ser montado e de pôr

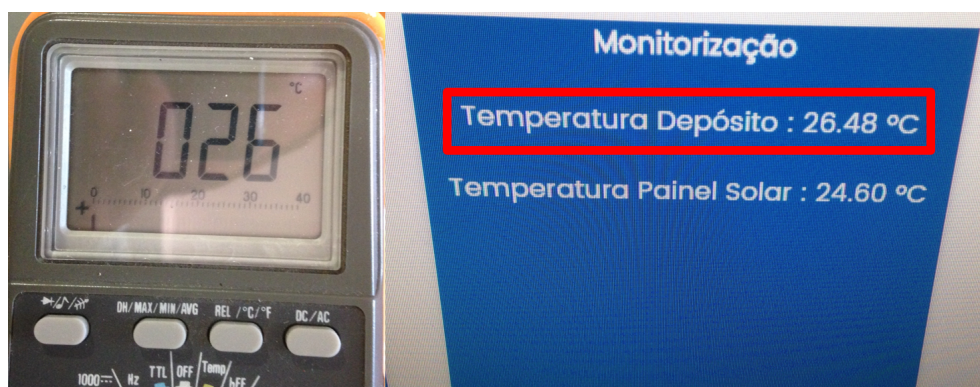


Figura 5.6: Comparação do valor medido com o LM35 conectado à placa ADS 1115 e com o sensor do multímetro.

em funcionamento, devido a envolver muitos componentes e sensores. Em termos de sensores os fins de curso funcionaram sempre tanto em modo automático como em modo manual. A detecção do sensor de fim de curso no sentido “trás” demora mais um pouco a parar o movimento devido a este ser outro *Esp8266* que não está conectado à placa de relés. A solução para acelerar a paragem poderá ser em tornar o código mais rápido, isto é, fazer a atualização de variáveis e a leitura das mesmas mais vezes. Em relação ao *Esp8266* conectado ao *encoder* este está a fazer a contagem correta dos impulsos e a realizar de forma correta o cálculo de quantos impulsos tem de contar para o tapete rodar o pedido pelo utilizador do sistema. No modo automático as paragens e as rotações dadas pelo sistema estão corretas, variando muito pouco do pedido, isto é, pedindo um intervalo de 10cm percorridos no tapete, este desloca-se uma distância de 11,5cm. Esta pequena variação poderá ser devido às rampas de aceleração e desaceleração realizadas pelo variador de frequência.

Na Figura 5.7 é possível observar o explicado anteriormente.

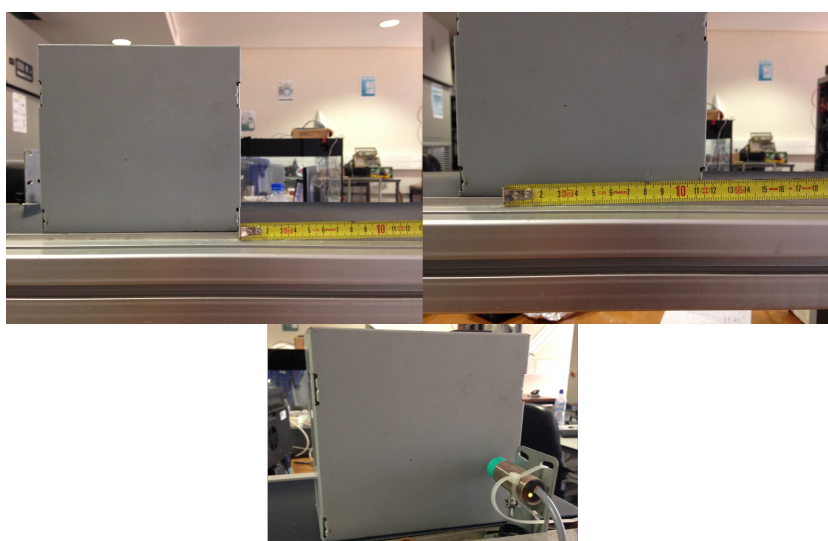


Figura 5.7: Resultados observados na utilização do sistema motor.

O último caso de estudo, **o sistema ar**, que tem como função controlar a temperatura de uma divisão de uma habitação, verificou um bom funcionamento geral. Os sensores de temperatura instalados no exterior e no interior do espaço controlado mediam valores exatos. Para se proceder a esta verificação usou-se o mesmo método que para o sistema 1, isto é, comparou-se o valor medido pelos sensores de temperatura, que é mostrado na interface, com o valor medido pelo termopar do multímetro. Nas Figuras 5.8 e 5.9 é possível observar essas comparações, primeiramente a comparação do sensor exterior e depois com o sensor interior.

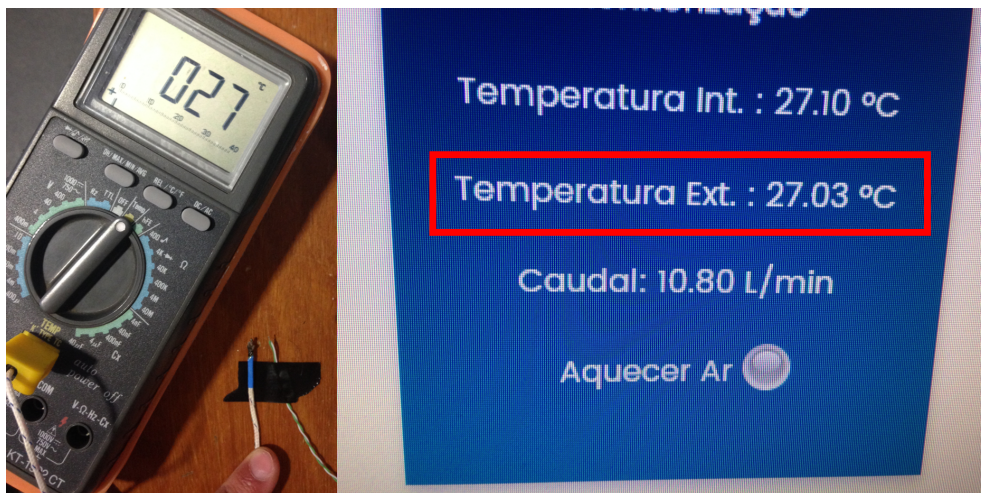


Figura 5.8: Comparação do valor medido com o termopar do tipo K conectado à placa ADS 1115 e com o sensor do multímetro.

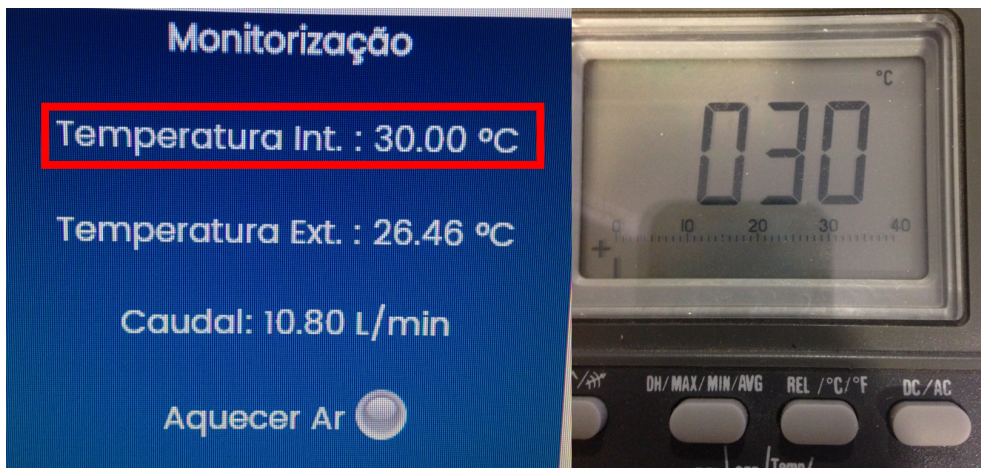


Figura 5.9: Comparação do valor medido com o LM35 conectado à ADC do *Esp8266* e com o sensor do multímetro.

Abordando os modos de funcionamento do sistema verifica-se que o modo manual funciona corretamente, sendo possível ao utilizador ativar e desativar as duas ventoinhas e o ventilador. O modo automático funciona bem também, cumprindo o intervalo de valores dado pelo utilizador e cumprindo a lógica mencionada e explicada anteriormente

em 4.3.4.

Em suma pode-se dizer que os sistemas propostos funcionam corretamente tanto no modo manual como no automático. Os sensores de temperatura retornam valores exatos, os sensores indutivos (de proximidade) retornam, também, a presença de forma precisa. As escolhas feitas pelo utilizador na interface são realizadas pelos sistemas e os dois modos, manual e automático, funcionam bem nos três sistemas. Deste modo é possível concluir que se os sistemas fossem implementados num caso real teriam forte probabilidade de funcionar corretamente e suprimir as necessidades do utilizador, no entanto certos aspetos ainda precisariam de ser melhorados. No Apêndice A pode ser visto com mais detalhe os circuitos dos sistemas montados.

No próximo capítulo serão tiradas as conclusões dos sistemas propostos, assim como será abordado quais poderiam ser os próximos passos para tornar o trabalho desenvolvido um produto viável.

Capítulo 6

Conclusão

O trabalho realizado no decorrer desta dissertação permitiu desenvolver vários sistemas para o controlo e monitorização de AQS, ar quente e o movimento de um painel solar. Destaca-se que os sistemas desenvolvidos são de baixo custo e podem ser importantes ajudas a sistemas de domótica.

Os sistemas desenvolvidos são baseados numa arquitetura Cliente/Servidor, constituídos por um computador servidor (**servidor**) e por vários *Esp8266* e respetivas placas com sensores e atuadores (**cliente controlador**). O Cliente e o Servidor interagem por *Wi-fi* através do protocolo 802.11, via TCP/IP. De notar que esta interação apenas acontece dentro da mesma LAN. Optou-se por esta abordagem, *Wi-fi*, devido a hoje em dia qualquer habitação possuir infraestrutura para tal e desse modo permitir uma maior flexibilidade ao utilizador de controlar e monitorizar os sistemas em qualquer ponto da habitação, outra motivo que também foi tido em conta na escolha do *Wi-fi* foi por assim os sistemas estarem um passo mais perto da “Internet das coisas” (*IOT - Internet of Things*). A interface de controlo e monitorização desenvolvida é compatível com dispositivos móveis ou fixos, deste modo é possível a partir de qualquer aparelho que esteja ligado à LAN da habitação controlar e monitorizar os sistemas em tempo real e visualizar um histórico de certos dados em forma gráfica.

Para o desenvolvimento dos sistemas foram usadas várias placas de desenvolvimento *Esp8266* MINI NODEMCU, acopladas a vários sensores de temperatura, LM35 e Termopar do tipo *K*, de presença, sensores indutivos, entre outros. Também foram usadas placas de relés acopladas ao *Esp8266* MINI NODEMCU para fazer a ativação de saídas. A escolha da placa *Esp8266* MINI NODEMCU deveu-se ao facto de esta ser possível de programar em linguagem Arduino e de já ter incorporada uma antena *Wi-fi*. De notar que durante o desenvolvimento do *hardware* foi sempre tido em conta o isolamento dos vários circuitos, para assim garantir o bom funcionamento de todos os componentes e a sua longevidade.

Durante o desenvolvimento de todos os sistemas foi sentida alguma dificuldade na realização dos circuitos, tendo vários erros, mas não danificando nenhum componente. Outro ponto em que se sentiu bastante dificuldade foi na criação dos casos de estudo para as análises de desempenho. Ao invés, durante toda a programação foi sentido algum à vontade, dado haver bases adquiridas em unidades curriculares anteriores.

Concluindo os sistemas desenvolvidos caracterizam-se, no global, por serem *user-friendly* e de baixo custo. Podendo ainda destacar que cumpriam todos os os objetivos em termos de grandezas a monitorizar e controlar.

6.1 Trabalho Futuro

Como referido anteriormente, os sistemas desenvolvidos cumprem com o que foi previamente definido, no entanto ainda haverá espaço a desenvolvimento. Destacam-se três aspetos importantes como parte de um trabalho futuro:

1. Construir placas de circuito impresso para todos os sistemas e acomodar essas mesmas placas em caixas específicas para o efeito. As placas podem ser desenvolvidas com recurso ao *software* EAGLE, e as caixas para acomodar as placas devem ser concebidas à medida para cada sistema, depois modeladas com auxílio de um *software* CAD (*Computer-aided Design*) e posteriormente impressas com recurso a uma impressora 3D;
2. Desenvolver um algoritmo que seja executado aquando do modo automático dos sistemas de controlo água e ar. Isto é, desenvolver um algoritmo que leia os valores da base de dados e depois ative as saídas conforme esses valores, além disso, este algoritmo deve ser capaz de “aprender” com base em valores de histórico. Deste modo, a eficiência dos sistemas iria aumentar, e a programação nos *Esp8266* seria mais padrão. Propõem-se ainda que este algoritmo fosse uma aplicação servidor;
3. Realizar testes aos sistemas em ambiente real. Os ambientes simulados em laboratório foram criados de modo a simular o mais possível o ambiente real, no entanto sempre que possível é melhor realizar a análise de desempenho em ambiente real devido a variáveis externas que não são tomadas em conta em laboratório.

Bibliografia

- [1] INE IP, DGEG. Inquerito ao consumo de energia doméstico 2010. Lisboa - Portugal; 2011. Available from: <https://www.ine.pt/ngt{ }server/attachfileu.jsp?look{ }parentBoui=127228533{&}att{ }display=n{&}att{ }download=y>.
- [2] Bergman TL, Lavine AS, Incropera FP, Dewitt DP. Fundamentals of heat and mass transfer. 7th ed. Jefferson City: John Wiley & Sons, Inc; 2011. Available from: <https://docs.google.com/file/d/0B3{ }fsg2ipzHca0NkUkpjcThFNnc/view>.
- [3] Pires AFdS. Desenvolvimento de automatismos e algoritmos de controlo inteligente, para monitorização e controlo de componentes habitacionais. Universidade de Aveiro; 2016.
- [4] CentralCasa. EuroX10 : X10Info;. Visitado em 2017-03-06. Available from: <http://www.eurox10.com/Content/X10Information.htm>.
- [5] WARMINGFIX LDA . Politéc - Assistência caldeiras e esquentadores e reparação de caldeiras; 2016. Available from: <https://politec.pt/>.
- [6] indústria metalúrgica lda A. Arierom; 2014. Visitado em 2017-03-09. Available from: <http://www.arierom.com/>.
- [7] Fundación Terra. Guía práctica de una instalación de energía solar térmica | Terra.org - Ecología práctica; 2007. Visitado em 2017-03-09. Available from: <http://www.terra.org/categorias/articulos/guia-practica-de-una-instalacion-de-energia-solar-termica>.
- [8] Horacio Zorrilla H. Arquitectura de Casas: Aire caliente central por bomba de calor.; 2015. Visitado em 2017-03-09. Available from: <http://blog.arquitecturadecasas.info/2010/05/aire-caliente.html>.
- [9] Alves. Termo-acumuladores com permutador de calor integrado | Canalizar e aquecer; 2013. Available from: <https://www.canalizareaquecer.com/esquentadores/termo-acumuladores-com-permutador-de-calor-integrado.html>.
- [10] PennWel. Renewable Energy World Magazine. Solar thermal water heating. 2012 feb;p. 50. Available from: <http://www.volker-quaschnig.de/articles/fundamentals4/index.php>.
- [11] LaMeres BJ, Nehrir MH, Gerez V. Controlling the average residential electric water heater power demand using fuzzy logic. Electric Power Systems Research. 1999 dec;52(3):267–271. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S037877969900022X>.

- [12] Henry's Bench. Arduino 4 Channel Relay Module | Henry's Bench;. Visitado em 2017-07-03. Available from: <http://henrysbench.capnfatz.com/henrys-bench/arduino-output-devices/5-volt-4-channel-arduino-relay-module-user-manual/>.
- [13] Microsoft. TCP/IP Protocol Architecture; 2017. Visitado em 2017-07-03. Available from: <https://technet.microsoft.com/en-us/library/cc958821.aspx>.
- [14] European Union. EU Transport in Figures - Statistical PocketBook 2016. 1st ed. Belgium, Bietlot: Luxembourg: Publications Office of the European Union, 2016; 2017. Available from: <https://ec.europa.eu/transport/sites/transport/files/pocketbook2016.pdf>.
- [15] Zadeh LA. Fuzzy logic. Computer. 1988;21(4):83–93. Available from: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=53>.
- [16] KNX Association. Mission and Objectives - KNX Association [Official website];. Visitado em 2017-03-02. Available from: <https://www.knx.org/knx-en/knx/association/mission-objectives/index.php>.
- [17] KNX Association. What is KNX? - KNX Association [Official website];. Visitado em 2017-03-02. Available from: <https://www.knx.org/knx-en/knx/association/what-is-knx/index.php>.
- [18] Ruta M, Scioscia F, Di Sciascio E, Loseto G. A semantic-based evolution of EIB Konnex protocol standard. 2011 IEEE International Conference on Mechatronics, ICM 2011 - Proceedings. 2011;p. 773–778.
- [19] X10. X10 Basics;. Visitado em 2017-03-06. Available from: <https://www.x10.com/x10-basics.html>.
- [20] Hepbasli A, Kalinci Y. A review of heat pump water heating systems. Renewable and Sustainable Energy Reviews. 2009 aug;13(6-7):1211–1229. Available from: <http://linkinghub.elsevier.com/retrieve/pii/S1364032108001032>.
- [21] Oluwatosin HS. Client-Server Model. IOSR Journal of Computer Engineering. 2016;16(1):67–71.
- [22] Rafiei M, Eftekhari SM. A practical smart metering using combination of power line communication (PLC) and WiFi protocols. Proceedings of 17th Conference on Electrical Power Distribution Networks (EPDC). 2012;p. 1–5.
- [23] Shenzhen Shun Technology Co . ESP-12F WiFi module; 2015. Visitado em 2017-05-25. Available from: <https://www.mikrocontroller.net/attachment/286085/ESP8266-12F{ }Tronixlabs{ }Australia.pdf>.

Apêndice A

Circuitos do Sistema Proposto

Os circuitos desenvolvidos para os vários sistemas durante a dissertação foram sempre construídos em *breadboard* para poderem ser feitos os devidos testes antes de se realizarem os protótipos de placa de circuito impresso.

Breadboard ou placa de ensaio é uma placa com furos e conexões condutoras dispostas de determinada orientação para facilitar a criação de circuitos elétricos e realizar os respectivos testes a esses. Estas placas facilitam bastante o desenvolvimento devido a não ser necessário soldar nenhum componente, dado que as pernas entram nos furos da placa.

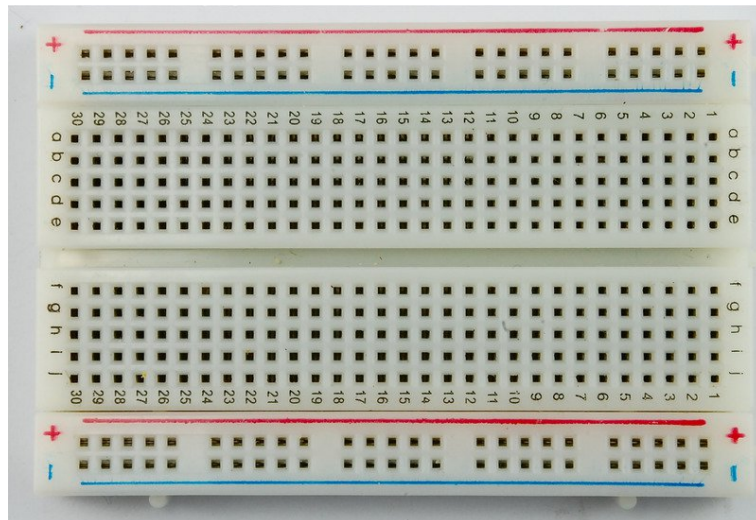


Figura A.1: *Breadboard* ou placa de ensaio.

Tal como visto no Capítulo 5 os circuitos foram montados em *breadboard*. Nas Figuras seguintes irão ser mostrados esquemas dos circuitos montados. Estes esquemas foram realizados com recurso ao *software* “Fritzing”. Este permite além de realizar os esquemas em *breadboard*, fazer o circuito elétrico e a placa de circuito impresso. De destacar, também, que este *software* é *open source*, o que faz com que não exista licenças de uso do mesmo e exista uma grande comunidade que faz peças que as bibliotecas não contém. Além disso lojas como a “Adafruit” e “SparkFun” disponibilizam bibliotecas dos itens que comercializam.

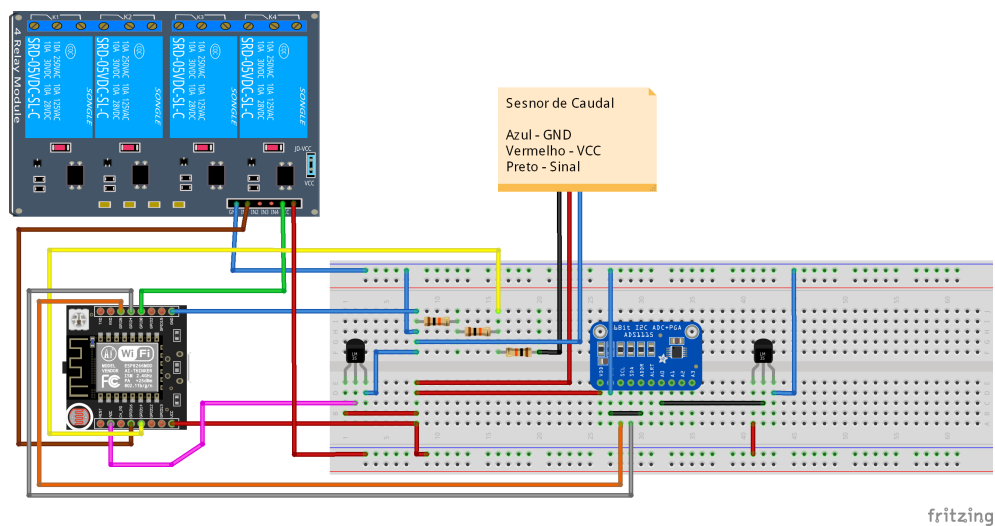


Figura A.2: Circuito do **sistema água** montado em *breadboard*.

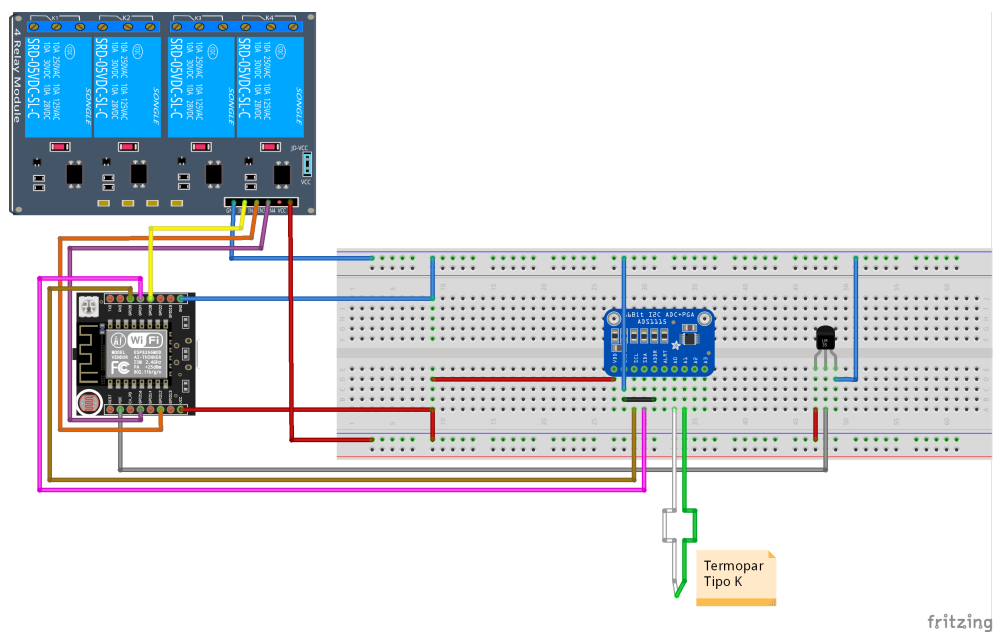


Figura A.3: Circuito do **sistema motor** montado em *breadboard*.

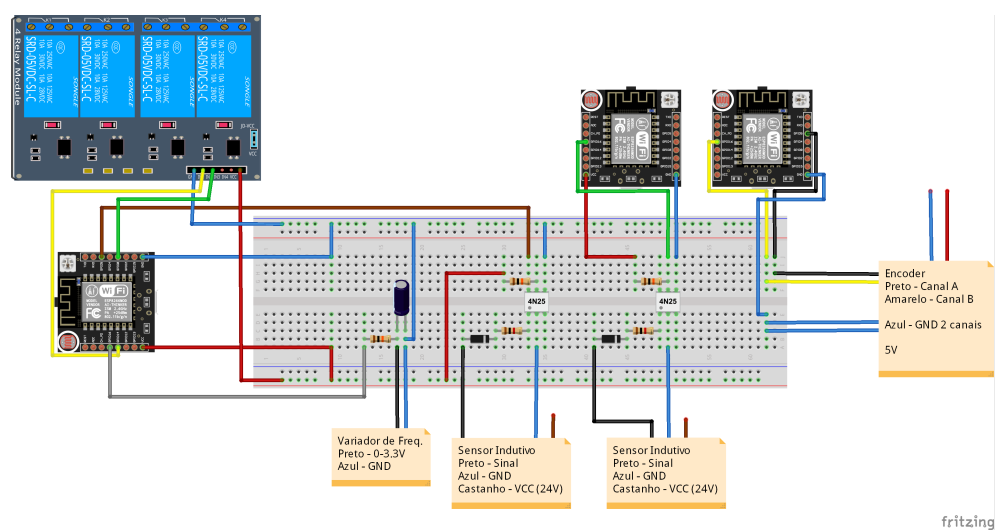


Figura A.4: Circuito do **sistema ar** montado em *breadboard*.

Apêndice B

Interface *web*

A interface *web* foi desenvolvida em linguagem HTML para poder ser utilizada em todos os equipamentos móveis e fixos que possuam um navegador *web*. Deste modo, como a maior parte dos dispositivos móveis e fixos contém um navegador *web* torna-se bastante útil e acessível que a interface tenha sido desenvolvida em linguagem HTML.

Além da linguagem HTML usou-se também vários *scripts* PHP e *JavaScript* para interagirem com a base de dados e também para fazerem atualizações em várias divisões da interface. Durante a realização da interface foi tido em conta o aspeto visual e a facilidade de interação assim como a compatibilidade entre ecrãs de várias dimensões.

B.1 Organização

A interface foi organizada em três principais áreas:

- Cabeçalho;
- Menu;
- Área de trabalho.

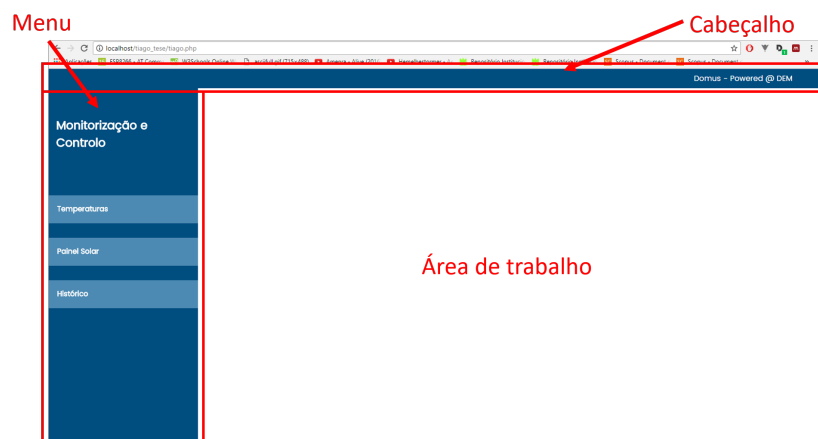


Figura B.1: Áreas principais da interface desenvolvida.

O Cabeçalho foi usado para fazer o topo da página e assim ter o nome da página e o crédito ao autor. Futuramente numa aplicação que dependa de utilizadores pode ser usado o cabeçalho para indicar o utilizador e até fazer o formulário de *login*.

O Menu é onde é possível alternar entre áreas de trabalho. Cada área de trabalho existente está marcada a azul mais claro e ao ser selecionada aparece na área de trabalho. Para a realização da interface criaram-se três opções, “Temperaturas”, “Painel Solar” e “Histórico”.

A Área de Trabalho é onde é possível fazer a monitorização e o controlo de todos os sistemas desenvolvidos. Ao clicar na opção pretendida no menu irá aparecer nesta zona várias possibilidades de interação com sistema.

B.1.1 Menu

O menu tem a característica de quando mostrado num ecrã em que não possa ter no mínimo 310 pixel de largura é colapsado para um botão que aparece no topo abaixo do cabeçalho. Este ao ser solicitado faz aparecer o menu sobre a área de trabalho, possibilitando assim ao utilizador escolher qual a opção que deseja. Na Figura B.2 é possível observar o dito anteriormente.



Figura B.2: Apresentação do menu em ecrãs de dimensões reduzidas.

B.1.2 Área de Trabalho

A área de trabalho apresenta o que está associado a cada uma das opções que estão no menu. Isto é, clicando sobre a opção “Temperaturas” irá aparecer o mostrado na Figura B.3. Nesta opção é possível controlar e monitorizar o sistema 1 e 3.

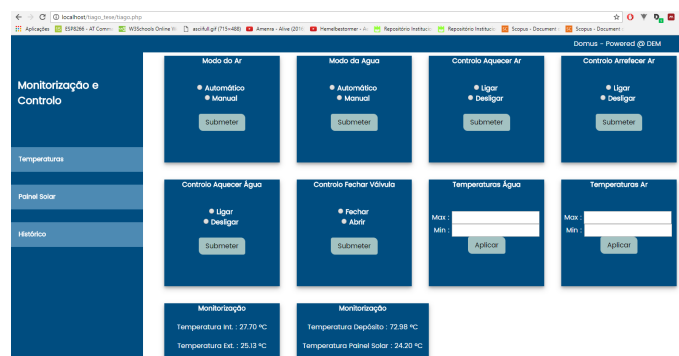


Figura B.3: Área de trabalho relativa à opção “Temperaturas”.

Fazendo o clique sobre a opção “Painel Solar” irá aparecer o mostrado na Figura B.4. De salientar que nesta opção a monitorização dos sensores e do sentido de rotação é dada pela variação de LED’s.

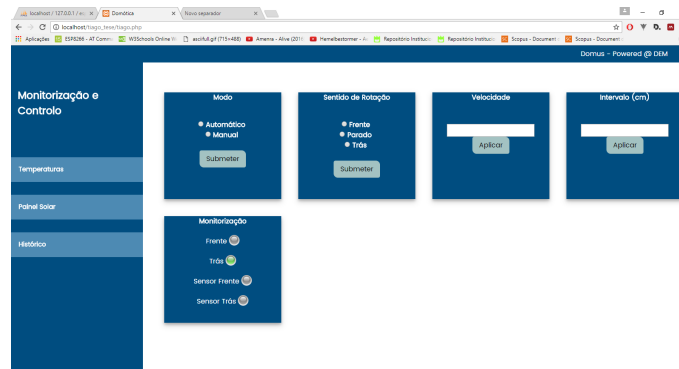


Figura B.4: Área de trabalho relativa à opção “Painel Solar”.

Entrando na opção “Histórico” é possível observar, através de um gráfico, um histórico de 50 valores da temperatura da água dentro do termoacumulador e da temperatura do ar dentro de uma divisão. Na Figura B.5 é possível observar o dito anteriormente. De notar que estes gráficos são sempre atualizados em tempo real.

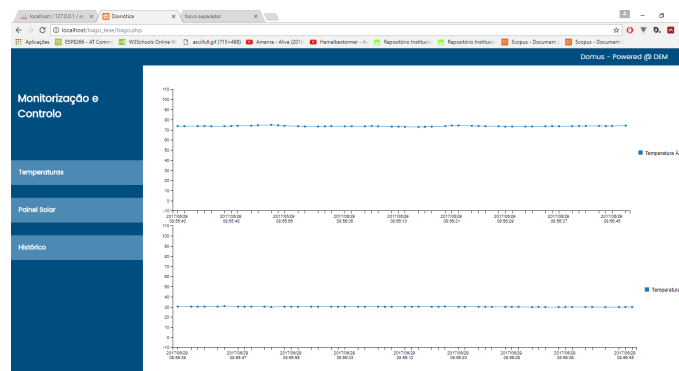


Figura B.5: Área de trabalho relativa à opção “Histórico”.

B.2 Considerações gerais

Para realizar a programação desta interface foi usado o *software* Notepad++, sendo que a interface está otimizada para funcionar no *browser* Google Chrome Versão 59.0.3071.115.

Para o desenvolvimento foram usadas dois ficheiros de estilo, ficheiros CSS (*Cascading Style Sheets*), . Estes ficheiros são provenientes da *W3 Schools* e da *C3js*. O ficheiro de estilo da *C3js* apenas é usado para a criação e geração dos gráficos. De notar que estes ficheiros de estilo são *open source*.